2017

Design of a Wearable fNIRS Neuroimaging Device with an Internet-of-Things Architecture

Gozde Cay
University of Rhode Island, gozde.cay@hotmail.com

DESIGN OF A WEARABLE FNIRS NEUROIMAGING DEVICE WITH AN INTERNET-OF-THINGS ARCHITECTURE

BY

GOZDE CAY

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF SCIENCE

IN

ELECTRICAL ENGINEERING

UNIVERSITY OF RHODE ISLAND

2017

MASTER OF SCIENCE THESIS

OF

GOZDE CAY

APPROVED:

Thesis Committee:

Major Professor Kunal Mankodiya

Walter Besio

James Agostinucci

Nasser H. Zawia
DEAN OF THE GRADUATE SCHOOL

UNIVERSITY OF RHODE ISLAND
2017

ABSTRACT

The brain is one of the most important and complex organs in the human body. It is responsible for the essential functions of the body. Brain activity monitoring is important in order the diagnosis of neurological disorders and for improving our understanding of the nervous system. To understand the structure and functionality of the brain, it must be monitored in natural or nonclinical settings such as during daily activities including finger tapping and thinking, as well as in outdoor activities like walking, jogging and cycling. This monitoring should assist in the detection of these natural circumstances. There are several techniques to monitor the brain's structure and function including functional magnetic resonance imaging (fMRI), electroencephalography (EEG), magnetoencephalography (MEG), positron emission tomography (PET), and single positron emission tomography (SPECT).

Functional near-infrared spectroscopy (fNIRS) is a non-invasive brain monitoring method which emits NIR light into the brain tissue and detects the transmitted or reflected light from the brain.

fNIRS has many advantages over its counterparts in terms of its relatively high temporal/spatial resolution, mobility, portability and safety. It has a higher temporal resolution than fMRI and a greater spatial resolution than EEG. The preparation for fNIRS measurement is also easier than for EEG, as conductive gel is not generally required. This also means patients do not need to wash their hair after fNIRS measurements. Moreover, it is safer compared to LASER technology due to the usage of NIR light.

Despite advances in fNIRS technologies, there remain challenges. Its physical set up, including both space requirements and the presence of long cables, contributes to the difficulty in using fNIRS systems for non-clinical environments. Preparation time and price are also concerns. For its use in non-clinical experiments, a portable fNIRS system must be wearable or easily carried by the subject. The cables must also be short and durable.

This master thesis research was aimed at addressing the hardware limitations of fNIRS by developing a wearable, small form factor fNIRS system. I have developed an fNIRS system based on the Intel Edison, a high-end embedded system designed for internet of things applications. The Edison on the fNIRS system drives the lighting and data collection process. The system is comprised of two components: a headband containing fNIRS sensors and a control unit which controls the data collection and transmission. The control unit includes an LED driver circuit to turn LEDs on and off, an analog to digital converter to digitize the analog value from the photodetectors and a microprocessor to control data collection and transmission. The control unit is connected to a computer via serial communication. The computer acts as a computing unit, storing the collected data and making the necessary evaluations and calculations. The thesis research involved developing three versions to achieve the goal of creating a robust, wearable, wireless system. We also optimized the intensity of emitted light, the sampling rate and communication between control unit and computing unit.

To develop a control unit, I first tested all the components on a breadboard. Then I combined the system on a single 6cm x 7.7cm PCB, effectively miniaturizing the

system. Then I connected the sensor with the computing unit and tested the system on a human to measure hemodynamic activity. Simultaneously data was collected from subjects' fingertips and arms for validation. In conclusion, the system successfully detected the subjects' pulses and hemodynamic changes.

Since the system is small enough to carry in a pocket or put in the back of the head cap, its next version will be a portable and wearable device with the potential to be used in ambulatory experiments.

ACKNOWLEDGMENTS

I would like to especially thank my advisor, Dr. Kunal Mankodiya, for his perfect guidance, and patience. I would also like to thank him for supplying me with an excellent research and working area. Without his guidance and advice, I would never have been able to finish my thesis.

I also would like to thank my committee members, Prof. Walter Besio and Prof. James Agostinucci, for accepting to be on my defense committee.

I would like to thank my colleagues at Wearable Biosensing Lab, especially Mohammadreza Abtahi and Manob Jyoti Saikia for being a family to me here and for their valuable help. Also I would like to thank Nick Constant and Alyssa Zisk for proofreading of my thesis.

Many thanks go to Turkish Government and Ministry of National Education for providing financial and spiritual support to me with a full scholarship during my master studies.

I also would like to thank National Science Foundation (NSF) to support our project financially.

Many thanks to my parents and my sister. Although they are not in the United States, they are always with me and support me.

Last but not least, I want to thank a lot to my dear husband, Mustafa Cay. He always encourages me even in difficult situations and supports me.

TABLE OF CONTENTS

LIST OF TABLES

TABLE PAGE

LIST OF FIGURES

FIGURE PAGE

CHAPTER 1

INTRODUCTION

The brain is one of the most complex organs in the human body. It controls essential functions such as thinking, feeling, storing memory, and coordination the body movements, actions and reactions. It is therefore of the utmost importance to keep the brain stable and healthy. However, brain disorders and brain injuries affect more people each year. For example 5.5 million Americans have Alzheimer's Dementia (AD) and this number is expected to grow rapidly through 2050 due to the fact that the population of Americans whose age range is 65 and older will increase from 48 million to 88 million [1]. Parkinson's disease (PD) is another common neurodegenerative disorder, currently affects about 60,000 Americans, more than 10 million people are diagnosed with it worldwide [2]. Almost 6,000 Americans have Amyotrophic Lateral Sclerosis (ALS) every year as well [3]. At least 2.5 million Americans experience traumatic brain injury (TBI) each year, with an unknown number of TBIs sustained by people who do not seek treatment. The Brain Injury Association of America notes that over 12 million Americans have acquired brain injuries [4].

Furthermore, according to the World Health Organization (WHO), neurological diseases constitute 12% of total deaths worldwide. Figure 1 shows the distribution of neurological diseases on number of deaths [5].

Cause category	2005 (%)	2015 (%)	2030 (%)
Epilepsy	0.22	0.21	0.19
Alzheimer and other dementias	0.73	0.81	0.92
Parkinson's disease	0.18	0.20	0.23
Multiple sclerosis	0.03	0.03	0.02
Migraine	0.00	0.00	0.00
Cerebrovascular disease	9.90	10.19	10.63
Poliomyelitis	0.00	0.00	0.00
Tetanus	0.33	0.23	0.13
Meningitis	0.26	0.17	0.10
Japanese encephalitis	0.02	0.01	0.01
Total	**11.67**	**11.84**	**12.22**

Figure 1 Percentage of deaths caused by neurological disorders in 2005, 2015 and 2030

Both Figure 1 and the discussion above help to make clear that neurological disorders and brain injuries have a major impact. Prevention, diagnosis, and treatment are of great importance. Therefore, continued or periodic brain monitoring should be done especially after diagnosis of a brain injury or mental disorder. This will provide additional insights to neurologists treating patients with these conditions.

1.1 Brain Monitoring Techniques

Neuroimaging or brain imaging could be defined as using imaging techniques to discover the states of neural systems in physiological and physical situations [6].

Neuroimaging could be divided into two categories:

- Structural neuroimaging is used for discovering the structure of the brain and neural system. It is also used to understand tumors and injuries which affect these structures.

- Functional neuroimaging is used to understand metabolic and connectivity changes in neural systems. It is used for brain-computer interfaces, when researching neurological and cognitive disorders, and to study some lesions as in Alzheimer's.

Table 1 compares neuroimaging systems based on their imaging type, measurement type, working principle, advantages, and disadvantages.

Table 1 Comparison of Brain Monitoring Systems

System	Imaging Type	Measuring Type	Working Principle	Advantages	Disadvantages
Computed Axial Tomography (CT)	Structural Imaging		Differential absorption of X-rays is used. X-ray absorbed by bone and hard tissues very well, little absorbed by soft tissues and very little absorbed by air and water. Therefore, CT scans pose the structure of brain but not in so much detail [7].	It poses gross features of brain, so it is good to diagnose fractures.	It does not supply very detailed brain image
Magnetic Resonance Imaging (MRI)			MRI machine is a large magnet whose magnetic field realigns the hydrogen atoms in patient's body. While these atoms come back their original statues, they produce radio frequency signal. The image of brain structure is created by detecting these signals.	Noninvasive, little health risk, provides stable imaging	Noisy and closed place is not comfortable for patients
Magnetoencephalography (MEG)	Functional Imaging	Electrical activity	Monitoring the magnetic fields which are naturally generated by neuronal activity [8].	Direct measurement of brain function, high temporal resolution, due to the fact that magnetic fields are generated naturally, it is safe	Low spatial resolution (compared to EEG), immobile, needs specialized shielding to eliminate the magnetic interference, non-invasive
Electroencephalography (EEG)			Electrodes are placed on the scalp and the electrical activity generated by neurons is measured from electrodes.	Mobile, high temporal resolution, non-invasive	Low spatial resolution, electrical conductivity—sometimes it is not certain which region of brain is active
Positron Emission Tomography (PET)		Hemodynamic and metabolic changes	Focusing on the level of the sugar glucose in the brain. Due to the fact that the active neurons use glucose as fuel, it checks glucose level to determine the location of neural firing happens. To monitor glucose level, it uses radioactive isotopes, short-lived radioactive material. When the isotopes are emitted, it creates visible spots which detector can detect [9]	It has the ability to recognize non-cancerous and cancerous tumors thus; it prevents patients from unnecessary surgeries, it can diagnose the neurological disorders like Alzheimer in early stage [10]	Low temporal resolution, immobile, does not allow continuous measurements because they use radioactive isotopes, only shows the generalized area, do not specify the location
Single Positron Emission Tomography (SPECT)			Works with the same procedure with PET. The main difference between SPECT and PET is that SPECT uses gamma rays instead of positrons [11]		
Functional Magnetic Resonance Imaging (fMRI)			Working with the same procedure with MRI, the main difference is fMRI focuses the hemodynamic changes in the brain. It measures the signal changes while the patient is performing mental tasks [9].	It is safer than CT and PET scan, has high resolution, for psychological evaluation, more objective	Low temporal resolution, immobile, physically constraining, susceptible to motion artifacts, exposes participants to loud noises, expensive, only look at the blood flow, do not localize the activity
Functional Near Infrared Spectroscopy (fNIRS)			Focusing the changes in the oxygen level in the blood. It measures the changes by emitting light to the brain tissue and detecting the reflected or transmitted data from the brain. If a region of the brain is active, there is more oxygenation in this area.	Since it uses light, it is safer than laser. It has relatively high temporal and spatial resolution. It can be portable. Inexpensive	Requires long cables between sensor and control unit. Table-top device so it is hard to make experiments in mobile environments. Shallow depth of penetration

Due to the fact that functional neuroimaging focuses on hemodynamic or metabolic changes in the brain, it is possible to diagnose abnormalities prior to the onset of structural changes. Precautions can then be taken in order to prevent or mitigate permanent damage due to neurological disorders or brain injuries. Therefore, this research is focused on functional neuroimaging. Although fMRI has a strong advantage due to its comparative safety and its high spatial resolution, it is limited by immobility, high cost and requirement for stillness. EEG, another functional neuroimaging technique, has the advantage of high temporal resolution and mobility. However; its low spatial resolution and the uncertainty of electrical conductivity remains a major disadvantage. As a solution of these drawbacks, functional near-infrared spectroscopy (fNIRS) is becoming more popular to monitor brain function [12].

1.2 Functional Near Infrared Spectroscopy

fNIRS is a non-invasive functional neuroimaging technique used for monitoring brain's hemodynamic activity [13][14].

The fNIRS system works by emitting light into the brain tissue and detecting the reflected light. fNIRS systems use NIR light ranging from approximately 700nm to 900nm in wavelength for two main reasons:

- NIR light passes through the scalp, the skull, and lipids with minimal absorption.

- Oxygenated Hemoglobin (HbO_2) and Deoxygenated Hemoglobin (Hb) are strong absorbers of NIR light. [15] Figure 2 shows the absorption spectra of Hb, HbO_2 and water [16].

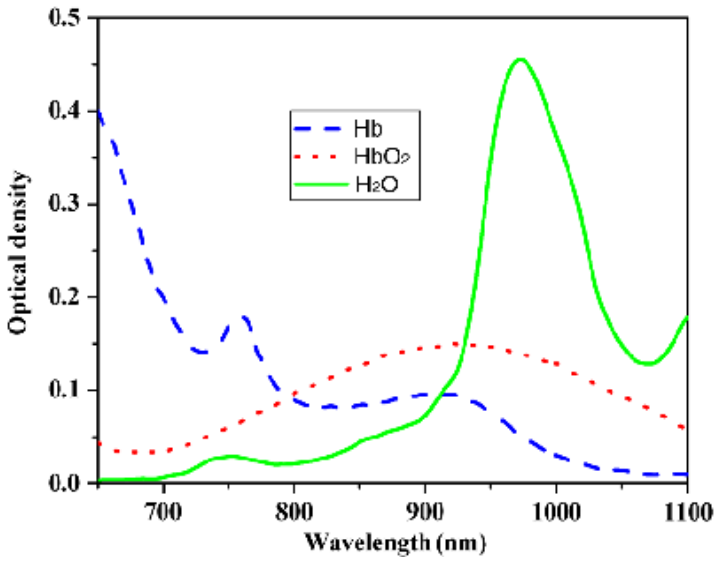


Figure 2 Absorption Spectra for HbO_2, Hb and H_2O[16]

According to the absorption spectra, Hb absorption peaks near a 700 nm wavelength, and HbO_2 near a 950 nm wavelength. To detect both HbO_2 and Hb, two different wavelengths close to these values must be emitted. For this reason, LEDs of at least two different wavelengths are used as sources in fNIRS systems.

Photodetectors are used to detect the reflected light. In order to fully cover the head for monitoring, a cap similar to EEG was used with the sources and detectors (also referred to as optodes) placed on it. As a note, optodes may also be placed on a headband to monitor the forehead alone.

The Modified Beer-Lambert Law is used to calculate absorption spectra of NIR light. The equations for Modified Beer-Lambert Law are shown in below:

$$\mu_\alpha(1) = ln(10)\varepsilon_{ox}(\lambda_1)C_{ox} + ln(10)\varepsilon_{de}(\lambda_1)C_{de} \quad (1)$$

$$\mu_\alpha(2) = ln(10)\varepsilon_{ox}(\lambda_2)C_{ox} + ln(10)\varepsilon_{de}(\lambda_2)C_{de} \quad (2)$$

The total concentration of Hb (C_{Hb}) is

$$C_{Hb} = C_{ox} + C_{de} \quad (3)$$

In these equations, ε_{ox} and ε_{de} represent the molar coefficients at the wavelengths λ_1 and λ_2. C_{ox} and C_{de} are the concentrations of the HbO_2 and Hb respectively (Wang, Wu, 2012).

1.3 The Aim of the Research

1.3.1 Problem

The fNIRS system has some advantages compared to other systems such as fMRI and EEG:

- Its temporal resolution (100 Hz or more) is greater than fMRI (10 Hz) [17],
- It does not require conductive gel like EEG,
- Since it uses NIR lights, it is safer than laser [18].

However, it still has several disadvantages:

- fNIRS systems require placing NIR sources and detectors on the scalp. Precise optode placement is critical to obtain high-quality fNIRS data.
- Each sensor and detector is connected to the system with long cables. These long, heavy affect patients' comfort and decrease the reliability of research.
- Commercial fNIRS systems are non-portable tabletop devices due to their large dimensions. Subjects must therefore go to the hospital and wait a long time for the devices to become available. Their dimensions also make experiments with movements such as walking nearly impossible.

- Although it is less expensive compared to the other systems such as fMRI, fNIRS remains costly for individual experiments.
- Hemodynamic response signals are acquired slowly, over the course of seconds.

1.3.2 Thesis Research on fNIRS Hardware Development

As a possible solution, we offer an Edison-based wireless and wearable fNIRS system. Our main goal is to develop a portable fNIRS system in order to make measurements in daily life. Brain activity can then be monitored while performing daily activities such as walking, jogging, driving etc. To do so, we miniaturized the fNIRS system by creating a small, inexpensive and more functional system using smaller component chips and reducing the number of wires. The comparison between the commercial fNIRS device and our implemented system is demonstrated in Figure 3.

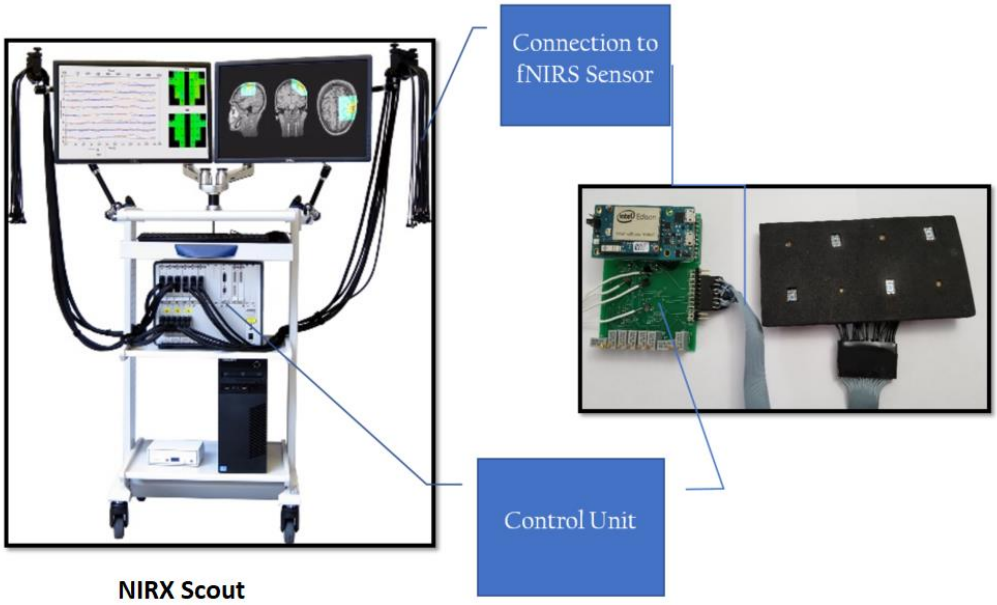


Figure 3 Comparison of commercial fNIRS device and our implemented system

The literature review showed few studies about the hardware aspect of fNIRS. The majority of the studies focused on the experimental side of fNIRS and validation of the advantages of fNIRS systems on clinical research, especially brain-computer interface studies. These studies also noted the continued limitations of fNIRS. The lack of experiments focused on hardware gave us motivation to study hardware development.

1.4 The Organization of Thesis

This thesis contains 4 further sections:

- The second section reviews the literature on fNIRS and other brain monitoring techniques.
- The third section evaluates the research and explains the methods to implement the research. I created three different fNIRS systems to achieve the goal of wearability and portability. Each system was explained in detail in this section.
- In the fourth section, the results of experiments made for verifying our system were shown. Fingertip pulse and hemodynamic changes during arterial occlusion were measured for this purpose.
- In the fifth section, the results were discussed. The problems faced during the development, implementation and experiments are explained. Concluding with suggestions for new solutions.

CHAPTER 2

REVIEW OF LITERATURE

This section will review the many types of research that have been conducted on both the experimental and hardware aspect of fNIRS.

First is the invention of in vivo fNIRS by Frans Jöbsis in 1977. He found that real-time non-invasive detection of hemoglobin (Hb) oxygenation was possible due to the relatively high degree of brain tissue transparency [19]. In 1984, David Delpy from University College London in the UK started NIRS measurements and represented the first quantitative measurement of hemodynamic parameters in sick newborn infants [20]. Then, in 1989, the first commercial system was released by Hamamatsu Photonics K.K. Companies continued developing NIRS prototypes and in 1994, the first 10-channel NIRS system was developed by Hitachi. The first simultaneous NIRS measurements with PET and fMRI were taken in 1995 and 1996, respectively. About a decade later, in 2009 Hitachi released a battery-operated, wearable, wireless 22 channel system to monitor the hemodynamic changes in an adult's prefrontal cortex. Figure 4 demonstrates the historical evolution of fNIRS system [15].

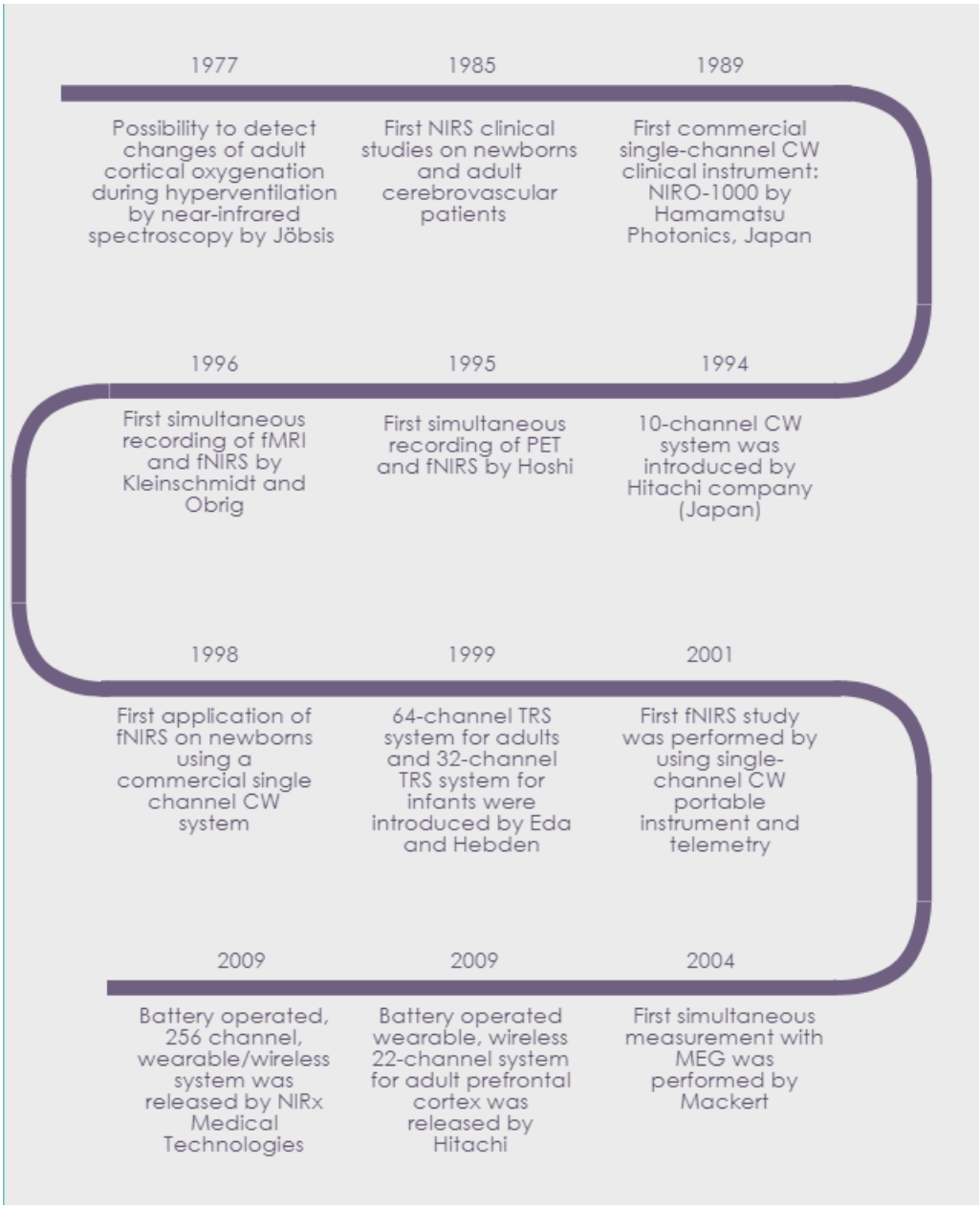


Figure 4 Historical evolution of fNIRS system [15]

2.1 Studies regarding Experimental Aspect of fNIRS

Most publications compare brain-monitoring systems such as EEG, MRI, fMRI, fNIRS, presenting fNIRS systems as a better solution.

In 2007, Coyle et al. offered the fNIRS system as a better option for BCI system. They made measurements using a “Mindswitch” which worked by moving their

hand/arm and by imagining moving their hand/arm. They claim the EEG-BCI system showed a good result for doing movements but the fNIRS-BCI system showed a better result for motor imaginary [21].

Also in 2007, Irani et al. did experimentally compared fNIRS and fMRI for neurological conditions like traumatic brain injuries, epilepsy, Alzheimer's disease, Parkinson's disease and psychiatric disorders like schizophrenia, mood disorders, and anxiety disorders. They also validated the use of fNIRS for these disorders. They showed fNIRS could be successfully deployed in clinical research and practice for these disorders. In addition, they reviewed the findings from studies on these disorders. [14].

In 2009, Lloyd-Fox et al. reviewed the history of fNIRS studies. They listed NIRS systems used in fNIRS studies and explained the development of fNIRS. They mostly focused on experiments for infants [22].

In 2011, Cui et al. compared NIRS and fMRI across multiple cognitive tasks in both temporal and spatial domains [23]. These tasks were composed of four experiments: left finger tapping, go/no-go, judgement of line orientation, N-back working memory task using visuospatial stimuli. Their experiments showed that NIRS signals have weaker SNR; however, it is correlated with fMRI signals. A correlation between BOLD response and a photon path developing an ellipse between the NIRS emitter was found in the spatial domain. Thus, it was proven that although NIRS could be a good alternative for fMRI, the spatial resolution and weaker SNR require careful examination. For NIRS data acquisition, they used ETG-4000 (Hitachi Medical, Japan) Optical Topography system. For fMRI measurement, they used 3T Signa Discovery 750 (GE Medical Systems).

All studies indicate fNIRS systems have advantages over MEG, EEG, PET, SPECT, fMRI, as well as limitations.

2.2 Studies regarding Hardware Aspect of fNIRS

In addition to research on the clinical and experimental application of Fnirs systems, there is work on the evolution of these systems.

In 2005, Bozkurt et al. developed a portable NIR system for monitoring newborns [24]. They indicated that newborns in neonatal intensive care units (NICU) are at a critical risk of brain injury and thus require continuous brain monitoring. They also stated that this monitoring system should be wireless in order to prevent interference with care. Bozkurt et al. offered a low-cost, battery operated, dual wavelength, continuous wave NIR system. It contained a probe for optodes, a control box and computer for data collection, signal processing and storage. The control box operated the light source and detectors. The computer controlled transmission and reception. The computer was also used for data storage and display. Two LEDs and one photodetector served as optodes, and they created a flexible circuit to connect the optodes. They used DAQcard 1200 as an analog to digital converter. Bozkurt et al. applied the hearing test to newborn babies and monitored the temporal region corresponding to the auditory cortex and the forehead for the control measurement. They also monitored electrical activity in response to sound with EEG. They claimed that there was an increase in both oxygenated and deoxygenated hemoglobin in the temporal region. Increase in blood volume indicates blood rushing to the local tissue. Also, the increase in deoxygenated hemoglobin was bigger than for oxygenated hemoglobin, showing local activity.

In 2006, Yurtsever et al. developed a wireless NIRS system [25]. They claimed that for receiving a robust signal, the probe should be comfortable, stable and provides good prevention from ambient light. To meet these requirements, they developed a flexible sensor pad. The flexibility of the pad allowed, the sensors to provide a good sensor-tissue coupling. They used four LEDs and ten photodetectors as optodes. Yurtsever et al. used a pocket PC with data acquisition card as a control unit. They provided a wireless communication between pocket PC and computer with the pocket PC as a client and the computer as a server. They used a cognitive optical brain imaging (COBI) studio software platform. For system evaluation, they used a tissue simulating phantom, and they stated that they received human-like results.

In 2007, Atsumori et al. offered a multi-channel, portable optical topography system [26]. According to their study, their system differs from previous optical topography systems is that they used vertical cavity surface emitting lasers (VCSELs) instead of optical fibers. By using this configuration, they eliminated the limitation on subjects' ranges of movement. They also claimed that they developed the first portable system to monitor the activity of the whole frontal cortex region. They used eight sources and eight detectors as optodes. They chose silicon photodiodes (Si-PDs) to prevent the head from heating. They used a field programmable gate array (FPGA) to process the data. The FPGA takes the signal and makes calculations on it, then saves it to flash memory. The FPGA and computer can communicate via both flash memory and wireless. To test their system, Atsumori et al. performed a word fluency task on one subject. They evaluated the changes in the concentrations of oxyhemoglobin and deoxyhemoglobin in working memory areas in the frontal cortex. They found that when

thinking started, the concentration of HbO_2 was increased and the concentration of Hb was decreased.

In 2010, Atsumori et al. developed a wearable optical topography system to monitor prefrontal activation during attention-demanding tasks performed while walking [27]. In this study, they aimed to use their previous wearable optical topography (WOT) system with OT measurements while walking. They used the same configuration (eight light sources, eight detectors and processing unit) with their previous WOT system. Rather than asking the subject to carry the processing unit as in prior studies, they placed the unit on the subject's wrist. They also asked subjects to perform an attention-demanding (AD) task instead of word fluency task. Like the previous study, communication between the processing unit and the computer could be performed either wirelessly or via flash disk. They aimed to monitor the activation in the dorsolateral prefrontal cortex (DLPFC) and the rostral prefrontal area in humans. They claimed that although they needed to investigate more subjects to generalize the data, their study showed that the WOT system was robust enough during walking to obtain task-related changes.

In 2013, Piper et al. developed a wearable fNIRS system for brain imaging in freely moving subjects [28]. Eight dual-wavelength LEDs were used as light sensors instead of optical fiber bundles, and eight silicon photodiodes were used as detectors. Each LED and detector was placed in a plastic housing and then mounted on a head cap. They designed a PCB for data processing and used it with a DAQ NI-USB 6216 card in an aluminum enclosure. They programmed a graphical user interface with LabVIEW 2011 on a small notebook. During experiments Piper et al. put the instrument and notebook

in a backpack. To evaluate the system, they performed self-paced left-hand gripping task with three different experimental conditions. These conditions are: (1) outdoor bicycle riding, (2) indoor pedalling on a training bicycle, and (3) sitting still. They aimed to monitor the primary motor areas of both hemispheres. To analyze the data, they used MATLAB. They reported that they had not seen any significant difference between pedalling and non-pedalling activities.

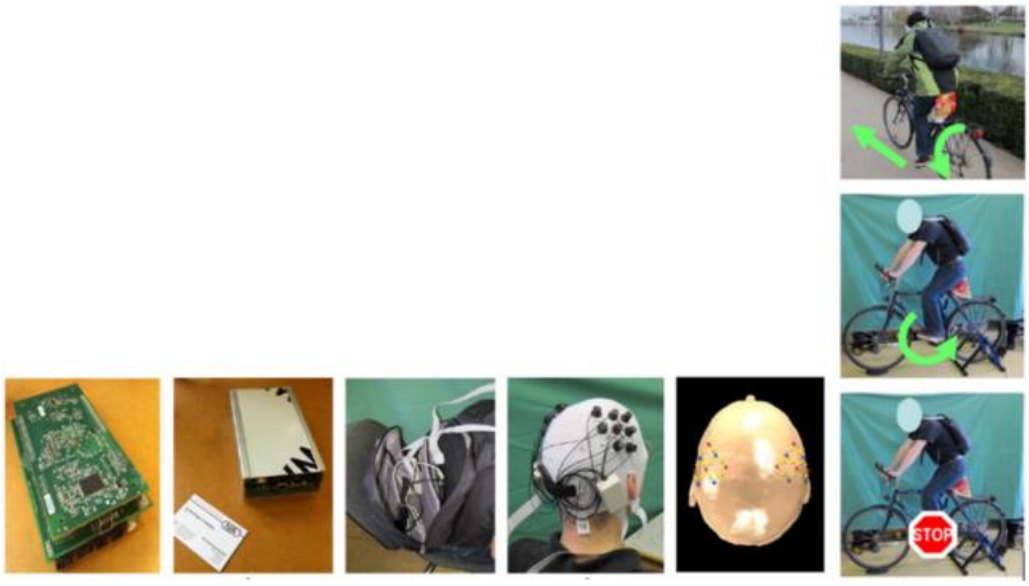


Figure 5 The wearable miniaturized NIRS system developed by Piper et al. [28]

In 2013, Zysett et al. developed a textile based fNIRS sensor [29]. They designed a patch from woven textile. To place the LEDs and photodetectors on the woven textile, they used flexible plastic strips. The photodetector strips are 4mm wide, and the LED strips 2mm wide. Strips were woven along the weft to connect them to the patch. Copper wires were used to connect the LEDs, detectors, and other components. They were interconnected with textile in the warp direction. They used two LEDs (Epigap ELC-760-26 – 760nm wavelength, Epigap ELC-870-17-2 – 870nm wavelength) in a strip and two strips, so four LEDs in total and two photodiodes (Vishay VBPW34S) with a

transimpedance amplifier. The contacts of LEDs, photodiodes and transimpedance amplifiers were covered with epoxy (Epo-Tek T7139) to prevent from breaking off. They then sewed the sensor into a textile cuff for use in measurements. As their experimental protocol, they set the LED on time to 1ms and LED off time to 1.5ms. The total timing for four LEDs is 10ms, for a frequency of 100Hz. They tested the sensor with a phantom first to measure noise. Then, they measured pulse waves from the fingertip to detect a heart rate. They detected a heart rate of 75 BPM and SpO_2 value as 94.1% ± 0.4%. Then, they performed venous occlusion with the textile sensor cuff. As an experimental procedure, they gave 2 minutes relax time first, then applied occlusion for 2 minutes, then released. Unlike in arterial occlusion, the oxygenation level increased during venous occlusion.

In 2015, Pinti et al. developed a fiberless, wearable fNIRS system for monitoring brain activity during cognitive tasks [30]. They designed a flexible probe unit consisting of six surface emitting laser diodes and six silicon photodiodes. They covered both the dorsolateral and rostral prefrontal cortex. Then they placed the fNIRS probes according to standard positions. They ran fNIRS system wirelessly with fNIRS acquisition software on a laptop. After they verified the signal quality, they turned the system off and ran stand-alone mode staring the data on the fNIRS system. After that, they arranged the cameras and started experiments. Three cameras recorded real-world tasks throughout. After the experiments were complete, they performed the data analysis with custom MATLAB scripts. They claimed that the results showed a hemodynamic trend typically related to functional activation.

Figure 6 Fiberless fNIRS system developed by Pinti et al. [30]

In 2016, Agro et al. developed a portable fNIRS system [31]. They aimed to cover the entire skull area, so they used sixty-four LEDs as light source and one hundred twenty-eight silicon photomultiplier (SiPM) optical diodes as detectors. They used SiPM diodes because the SiPMs shows an ideal SNR without any cooling system. They divided optodes into flexible boards; each board consists of four LEDs and sixteen SiPMs. An Advanced RISC Machines (ARM) microcontroller based main board was designed for LED switching and the acquisition from the SiPM. UART/RS232 and UART/USB interfaces handled communication between the main board and the computer. To evaluate the system, they performed a breath holding experiment with a volunteer subject. The result of the experiment showed a significant variation of the total hemoglobin and they claimed that it indicated that the grey matter of the brain had been reached.

CHAPTER 3

METHODOLOGY

Portable fNIRS systems have to meet certain requirements in order to make reliable and accurate measurements [32]. These requirements are detailed below:

- fNIRS measurements can occur over a long period of time, so the comfort of participants is critical. Participants have to relax in order to get accurate data. For this reason, the headband or head cap which carries the fNIRS sensors must be as light as possible.
- The headband or head cap must also be tight and flexible enough to keep the optodes stable on the scalp. Having stable contact with the skin is critical for collecting reliable data.
- Moreover, the system needs a powerful microprocessor to control optodes, sending light pulses, and collect the data. For wearable system, the system should use a rechargeable battery.

The master thesis was focused to develop the fNIRS system with these requirements in mind. The fNIRS system contains three main components: an fNIRS sensor, a control unit, and a computing unit. The fNIRS sensor consists of photodetectors and LED emitters. The control unit sends signals to command the LEDs and collect the data from photodetectors. It connects to the fNIRS sensor via wires. The computing unit runs the signal processing and data processing software. The computing unit connects to the control unit either via serial communication or wireless

communication. Figure 7 shows the overall block diagram of the fNIRS system developed for this research.

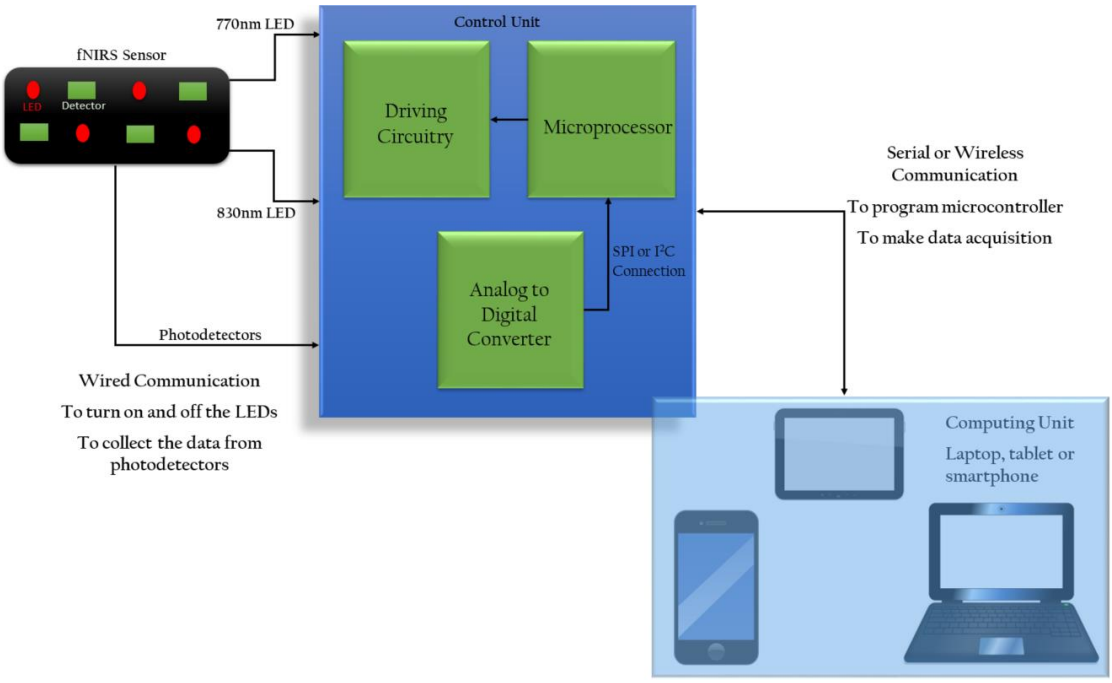


Figure 7 An Overall Block Diagram of the fNIRS System

For my research, three different fNIRS systems were designed–Each new version was evolved from the previous one to address the challenges incrementally. We started with a basic system containing a microcontroller placed on a headband along with LEDs and photodetectors. It met the requirements of portable systems. The headband was made from a flexible and soft material, so it fit the subjects' heads comfortably and tightly. A 3.7V rechargeable battery allowed the system to operate for up to 6 hours. However, the microcontroller's output voltage was insufficient to drive the LEDs strongly enough and brain signal quality was poor. For this reason, we switched to another microcontroller which can supply more powerful output. Also, an LED driver circuit was designed to trigger the LEDs. Although this system was more powerful, it

was too big to be portable. We decided to miniaturize the system by using only the necessary chips and components. Figure 8 shows the evaluation of the study.

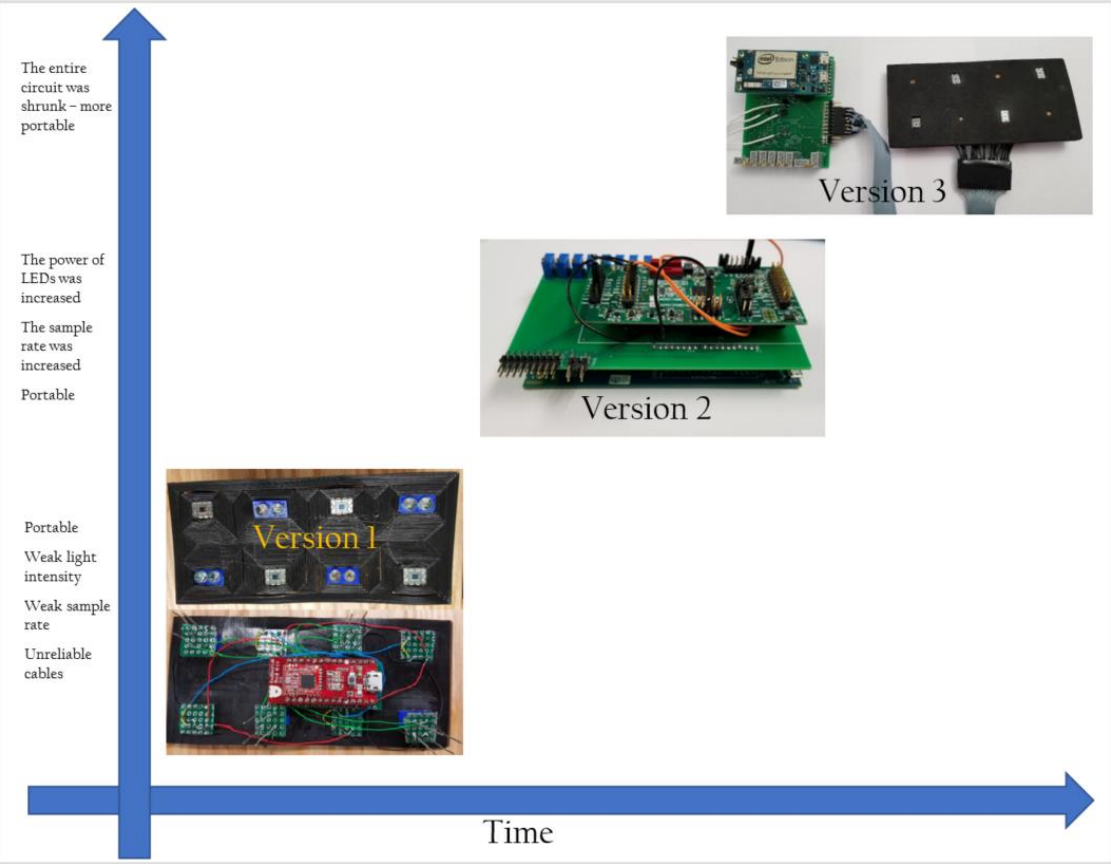


Figure 8 Evolution of the fNIRS hardware development within this thesis study

3.1. Version 1: Flexible Headband and Blend Micro

In our first experiment, a semi-flexible patch was designed to carry the LEDs, the detectors, and microprocessor. The patch was placed on the forehead with a headband. Figure 9 shows the block diagram of the system [33].

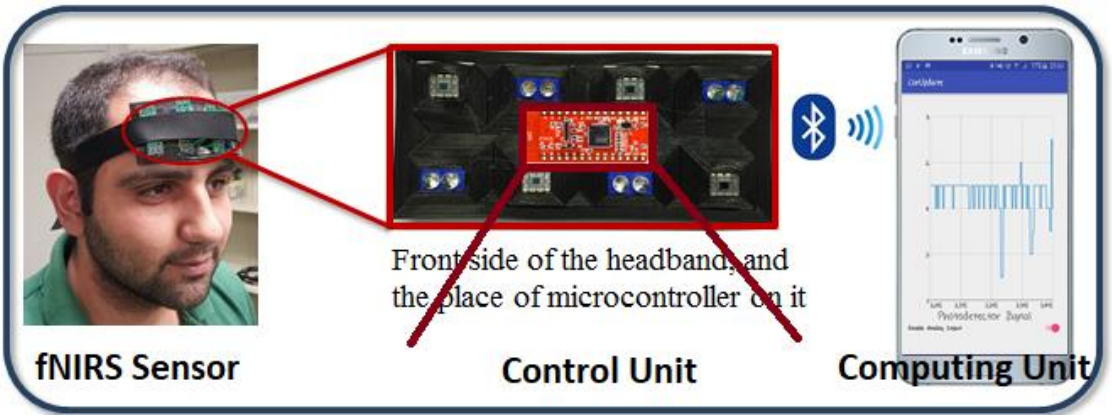


Figure 9 A Block Diagram of Implemented System

The hardware includes 8 LEDs and 4 photodetectors for detecting the signal from the forehead. We used two types of 5mm single wavelength LEDs. Their wavelengths were 770nm and 830nm. The photodetector used was a monolithic photodiode with a single supply transimpedance amplifier (OPT 101, Texas Instruments). A 3D printed flexible headband held the optodes and microcontroller in place. Thus, the system was completely wearable. All the wires between the optodes and the microcontroller were hidden in the flexible headband. The system was run off a 3.7V rechargeable Lithium-ion battery. We used Blend Micro as a microprocessor. The digital I/O pin of Blend Micro was used to trigger the LEDs, and the analog inputs were used to collect the analog data coming from photodetectors. Figure 10 shows the algorithm used for triggering the LEDs.

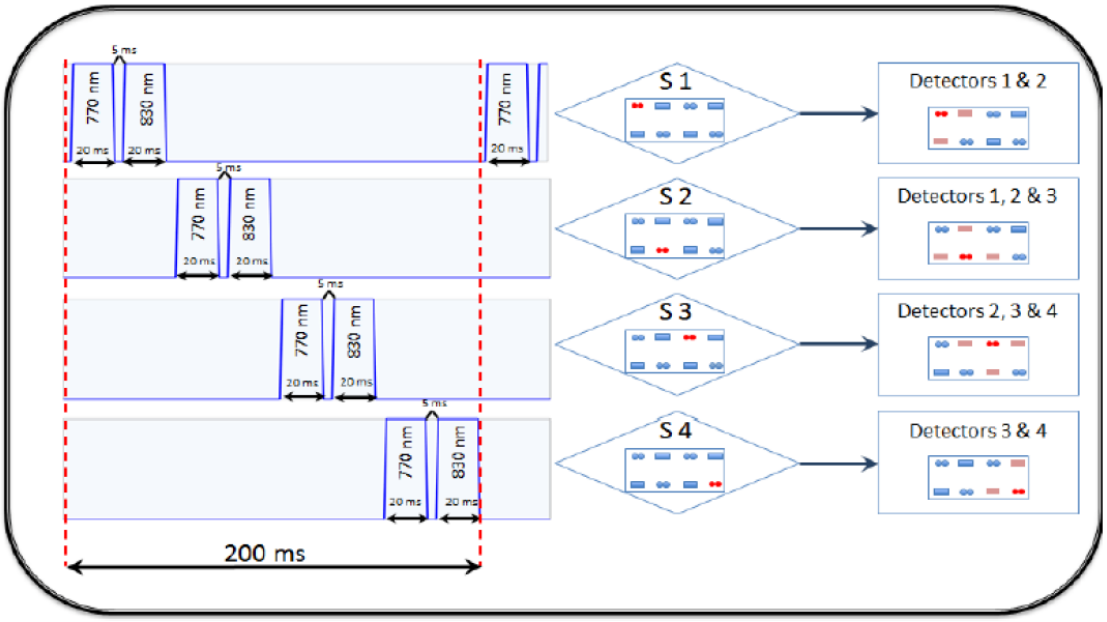


Figure 10 The timing diagram of triggering LEDs and collecting the signal

As can be seen from the algorithm, each LED turns on for 20ms and turns off for 5ms. The total timing is therefore 200ms, which corresponds with a 5Hz frequency. To collect accurate data, we considered the signal coming from nearby detectors. The active detectors are shown in red color. For example, when S1 (LED pair 1) was triggered, Detector 1 and Detector 2 were the most active. In this design, the distance between sources and detectors was 3 cm. This montage included 4 pairs of LED and 4 photodetectors. Each wavelength provided 10-channel data, providing 20 total measurements per frame. Figure 11 shows the signals collected from a LED and photodetector pair in a dark environment.

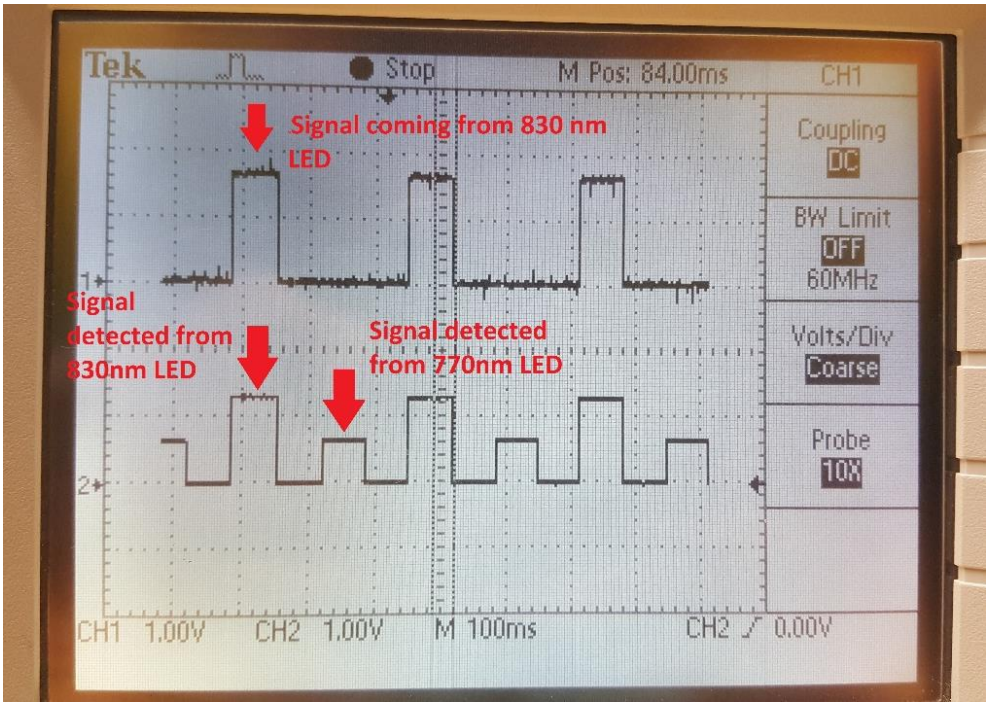


Figure 11 Signals from LED and photodetector

Although this system was small and compact enough to be a wearable wireless system, it still had disadvantages regarding comfort, power, and wireless communication. The material used to 3D-print the patch was not comfortable for long wear. Also, using two separate LEDs for the two wavelengths consumed much space on the band, increasing the size of the band. As control unit, we used Blend Micro Embedded Board and its microcontroller (Atmega32U4 Microcontroller) to trigger the LEDs. We collected accurate signals from the arm, but the reflected signal could not be detected from the forehead. I therefore concluded that the light emitted was too weak, as it was completely absorbed or scattered. The output voltage of Blend Micro digital I/O pins is 3.3V, which triggered the LEDs at an insufficient intensity. Moreover, the Blend Micro built-in analog to digital converter only had a 10 bits sampling rate. This resolution was not enough for fNIRS measurement. For wireless communication, we used Bluetooth (IEEE 802.15.1) communication protocol in this design. Bluetooth has

low power consumption, so it is good for battery-sourced devices, but it has limited sampling rate and frequency bandwidth for real-time fNIRS measurement.

3.2 Version 2: Sandwich Model

By considering the limitations of our previous design, we reviewed the possible embedded boards, wireless communication techniques and possible components. Table 2 shows the comparison of possible embedded boards.

Table 2 Comparison of embedded boards based on their technical features

	Board	Processor	Speed	RAM	Memory	Ethernet	Wi-Fi	Bluetooth	Power
1	BLE Nano	ARM Cortex-M0 SoC	16 MHz	32 kB/16 kB	256 kB/128 kB	No	No	BT 4.0 Low - Energy	3.3V
2	Blend Micro	Atmel ATmega32U4	8 - 16 MHz	2.5 kB	32 kB	No	No	BT 4.0 Low - Energy	5V
3	Intel Edison	32 bit dual-core Intel Atom SoC + Intel Quark	500 MHz + 100 MHz	1 GB	4 GB	No	Dual band (2.4 and 5 GHz) IEEE 802.11a/b/g/n	BT 4.0 Low - Energy	7-15V (9V)
4	Intel Galileo	Intel Quark SoC x1000	400 MHz	256 MB	8 MB	10/100 Mbps Ethernet *RJ45 port	No	No	12V
5	Raspberry Pi 3	64-bit quad core ARMv8	1.2 GHz	1 GB	No internal storage	Yes	IEEE 802.11n	BT 4.1 Low - Energy	5V
6	Particle Photon	ARM Cortex-M3	120 MHz	128 kB	1 MB	No	2.4 GHz 802.11b/g/n	No	5V

Since Blend Micro has limited sampling rate and frequency bandwidth, we needed to replace it with more powerful and faster embedded board. We decided to use the Intel® Edison Compute Module because of its speed and RAM capacity. Table 3 shows the comparison of wireless communication techniques.

Table 3 Comparison of Wireless Communication Techniques

	Protocol	Range	Data Rate	Frequency Bandwidth
1	ZigBee (IEEE 802.15.4)	10-100 m	250 kbps	2.4 GHz
2	Bluetooth (IEEE 802.15.1)	1-10 m	1-3 Mbps	2.4 GHz
3	WLAN (IEEE 802.11a/b/g/n)	35-120 m / 70-250 m	54-150 Mbps	2.4, 3.7 and 5 GHz

We need faster communication between the device and the computer and/or mobile device, so we decided to use Intel® Edison's Wi-Fi technology instead of Blend Micro's Bluetooth technology. Also, to supply enough power to LEDs, we decided to design an LED Driver Circuit. Safety assessments for NIR light intensities and tissue heating were performed by Cope 1991, Strangman 2002 and Bozkurt 2004 [34][12][35][36]. Standards for lasers, IEC68025, IEC62471 are used as adjustment because there are no safety standards for single wavelengths in LEDs. The major effect of LED is tissue heating. While adjusting the light intensity, we considered the maximum permissible exposure (MPS) of NIR light. According the Strangman, the MPS ranges from 0.2 W/cm^2 at 630 nm to $0.4W/cm^2$ at 850 nm. We applied the amount of light below these points and we checked the reflected light detected by the photodetectors. Comparing these signals with pre-verified signals, we tried to determine the enough intensity. The block diagram of the system is demonstrated in Figure 12.

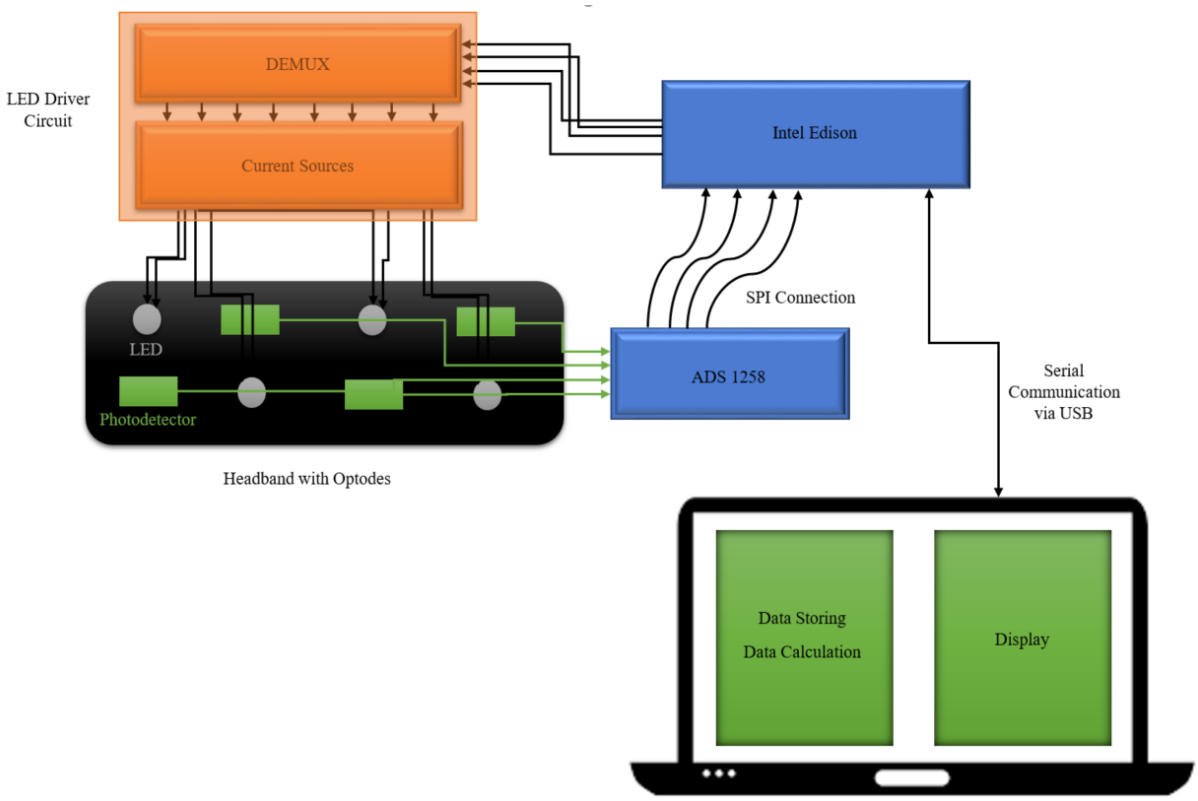


Figure 12 A Block Diagram of Version 2

3.2.1 fNIRS Sensor: Flexible Headband

We designed another flexible headband by using sponge/foam material, and 3D printed flexible circuit. We chose sponge/foam material for flexibility and to match the curve of the forehead. We designed the circuit of optodes in Eagle and used a milling machine (OtherMill^{PRO}) to print it on the flexible sheet. Then, we manually soldered the LEDs and detectors. In this way, we ensured flexibility for both the headband and the optodes circuit. Figure 13 demonstrates the design of headband and flex circuit.

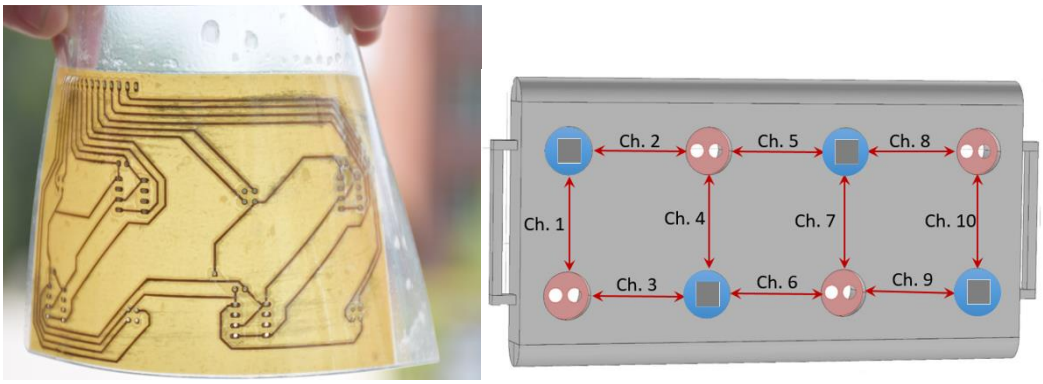
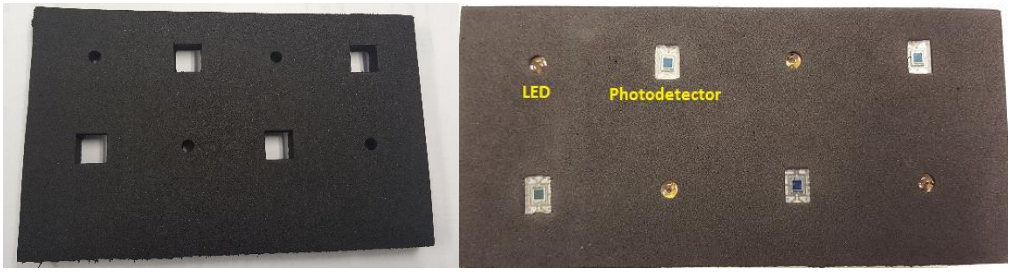


Figure 13 Flex Circuit for Optodes

As we discussed before, the patient comfort is vital to collecting reliable and accurate data. For this reason, the headband worn during experimentation should be soft and light. To ensure comfort, we placed the flexible circuit into a foam headband. The foam headband also kept the flexible circuit tight, establishing and maintaining good contact between the optodes and the skin. It is also important to emit an intense enough light into the brain to collect a strong reflected signal. Figure 14 shows the foam headband and its design.



(a) (b)

Figure 14 (a) Foam headband for flexible circuit (b) Headband with flex circuit

Our hardware included 4 LEDs and 4 detectors. To make the system more robust and look nicer, we decided to use multi-wavelength LEDs instead of LED pairs. Each LED can emit three wavelengths: 770nm, 810nm, and 850nm. We used the 770nm and 850nm wavelengths. The maximum current level in the LEDs is 200mA.

3.2.2 Control Unit 1: LED Driver Circuit

To emit enough light to brain tissues, we need to send enough power to trigger the LEDs. To do so, we built a LED driver circuit.

We used programmable current source (LT3092 – Linear Technology [37]) to supply enough current to the LEDs. The LT3092 provides up to 200mA output current, so it is powerful enough to trigger the LEDs at the required intensity. It requires two resistors to supply this current, a fixed resistor and a potentiometer to adjust the amount of current. We used a 1Ω fixed resistor and a 20kΩ potentiometer to achieve 200mA current.

Since we used multi-wavelength LEDs, we needed eight current sources for 4 LEDs. This increased the number of wires between the Intel Edison and LED drivers. Each of the drivers also had to be programmed individually which increased the complexity of the programming code. To reduce the number of wires and simplify the program we used a demultiplexer (DEMUX) (CD74HC4067 – Texas Instruments [38]) The common output of the fixed resistor and potentiometer was connected to the DEMUX output pin. DEMUX input/source pins were connected to the Intel Edison Digital I/O pins. When the Intel Edison sends the triggering signal, the DEMUX is configured according to its truth table and switches the outputs. The current sources received the signal from the DEMUX and triggered the LEDs according to their current value. Figure 15 shows the design configuration.

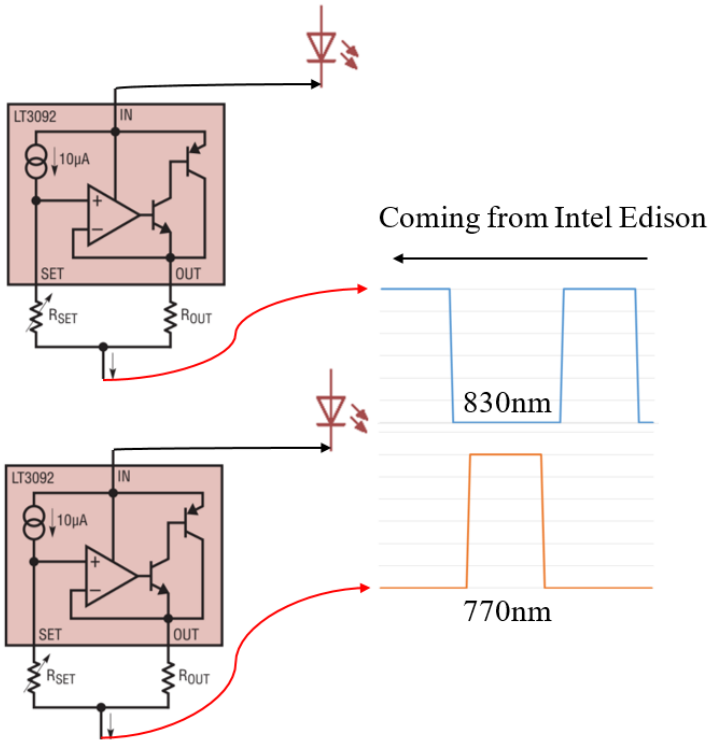


Figure 15 Design configuration for LED Driver Circuit

The truth table and working procedure of DEMUX are explained in detail in Appendix E.

The schematic configuration of LED Driver Circuit is provided in Appendix F.

3.2.3 Control Unit 2: Analog-to-Digital Converter

The ADC chip embedded on Blend Micro that we used in the previous study had insufficient resolution. It was 16 channel 10 bits ADC chip and it can drive up to 16MHz. Real-time fNIRS measurements require higher sampling rates. Therefore, we decided to use an external ADC chip.

We need high speed and high data rate so we decided to use an ADS1258 Analog-to-Digital Converter (Texas Instruments[39]). Its features are:

- 24 bits
- Fixed-channel data rate: 125kSPS

- Auto-scan data rate: 23.7kSPS/Channel
- 32.768kHz Crystal Oscillator or External Clock
- 16 single-ended or 8 differential inputs [39]

Figure 16 illustrates the ADC Evaluation Board.

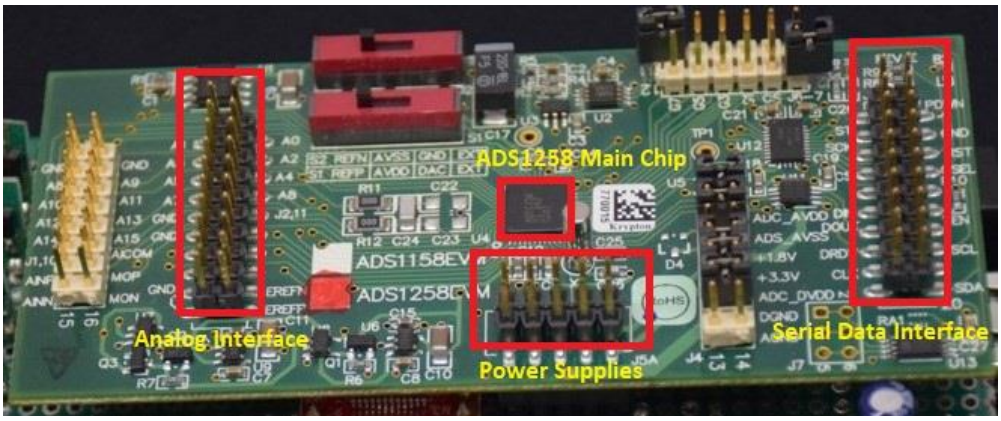


Figure 16 ADS1258 Evaluation Board

The photodetectors send the detected signal from the brain to the ADC. The ADC chip converts this analog signal to digital signal and sends it to the Intel® Edison via Serial Peripheral Interface (SPI) connection.

3.2.4 Control Unit 3: Intel® Edison Arduino Board

The Intel® Edison module is a SoC (System on Chip) board. Its key features are:

- Intel® Atom™ 500 MHz dual-core, dual-threated CPU,
- Intel® Quark™ 100 MHz microcontroller
- Integrated Wi-Fi, Bluetooth 4.0 LE

The Intel® Edison Kit for Arduino provides the Arduino 1.0 pinout. For communication, it provides standard connectors such as a micro USB connected to a UART, a USB On-the-Go (OTG) port that can be switched between a second micro USB device connector, a standard size USB host Type-A connector, a µSD card holder, and a DC power jack. It provides 20 digital input/output pins, of which 6 can be used as

analog inputs. The digital IOs and analog pins can be configured to operate at either 5V or 3.3V. The outputs can act as sources or sinks for 24 mA at 3.3V and 32 mA at 5V [40].

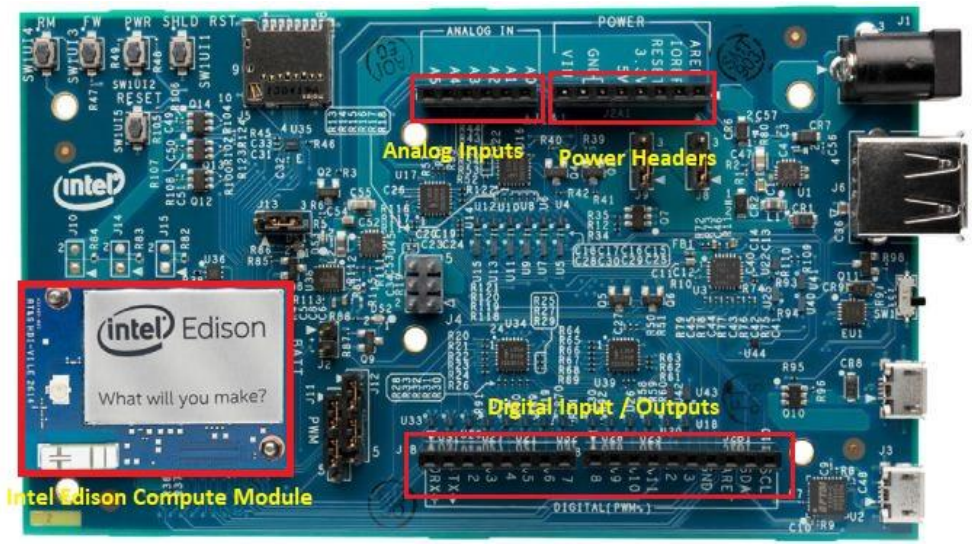


Figure 17 Intel Edison Arduino Board

Since it supports 32 mA at 5V, we decided Intel® Edison Arduino Board is a suitable choice for collecting a strong signal.

We programmed Edison's microcontroller to trigger the LEDs at defined time, to receive digitized photodetector data from the ADC, to store the values, and send the values to the computing unit wirelessly or via serial. To make the system small and wearable, we sandwiched these three circuits. Figure 18 shows the sandwiched system.

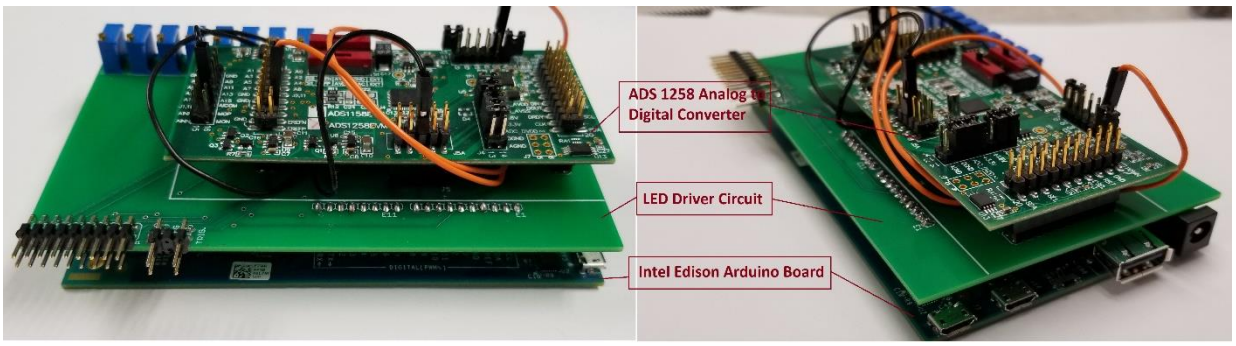


Figure 18 Sandwiched System

The whole system was put in a 3D printed box. Figure 19 shows the boxed system.

Figure 19 The boxed wearable fNIRS system

There are two ways to power the system.

1) Connect the middle micro USB port to the computer. This power is enough to upload the sketches to Edison and make basic commands.

2) Connect a 9V adapter or battery with a battery holder to power jack of Edison. This power is enough to perform fNIRS commands.

3.2.5 Computing Unit: Software

A control algorithm was written in the Arduino IDE. The Arduino IDE is an open-source software platform developed by Arduino to write code and upload it easily to any Arduino board [41]. It uses serial communication to identify the board and upload the code. The programming language is based on C. Intel Edison Compute Module is compatible with Arduino boards, so we were able to use the Arduino IDE without any problem. We programmed the Intel Edison to generate eight different digital signals to control the DEMUX and thus trigger the LEDs. The SPI pins of the Intel Edison received the digitized signal for further processing. To visualize the data, we programmed a Matlab (Mathworks Inc. [42].) graphical user interface (GUI). In this

GUI, we were able to visualize the raw data from channels and the level of oxyhemoglobin and deoxyhemoglobin as well. The code to collect raw data from the fNIRS sensor is provided in Appendix C.

3.3 Version 3: Wearable fNIRS (W-fNIRS)

Although the sandwiched system is compact and wireless, it is still too big for participants to carry. Therefore, we decided to miniaturize the system by not using the daughter boards. The difference between Version 2 and Version 3 is demonstrated in Figure 20.

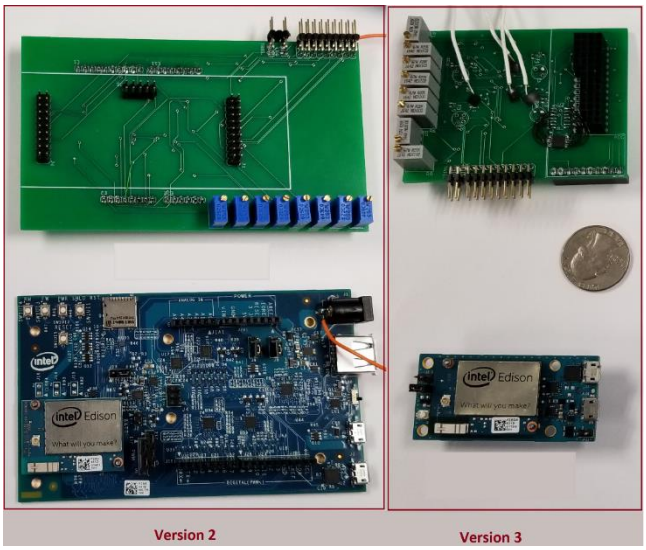


Figure 20 The difference between Version 2 and Version 3

This third version of the system was again composed of three main parts. These parts are an fNIRS sensor headband, an LED driver circuit, an analog to digital converter, and the Intel Edison as the control unit and software on the computer as computing unit. Figure 21 shows the block diagram of the circuit.

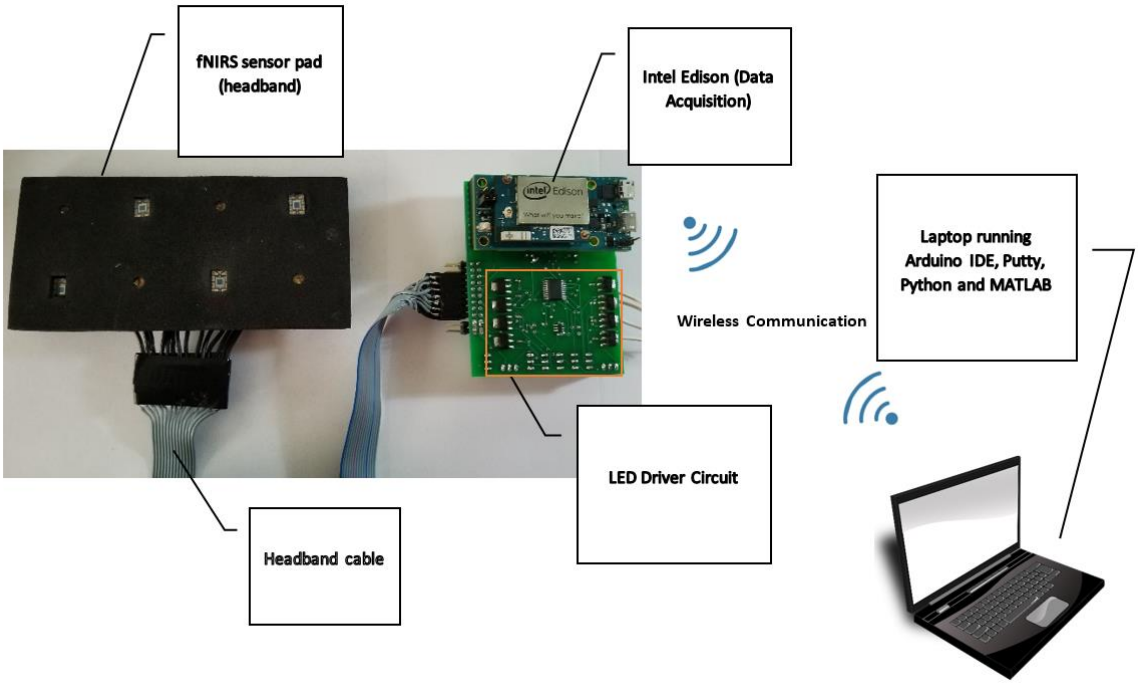


Figure 21 Block diagram of system

These parts of the system are explained below.

3.3.1 fNIRS Sensor: Headband and Optodes Design

In this part of the study, we used both the flexible headband and a head cap which carries the optodes. The head cap allows us to place the optodes on different regions of the head, rather than being limited to the forehead. The detectors were also changed. Due to the fact need for better sensitivity and response, we used 940 nm photodiodes (ODA-6WB-500M, Opto Diode Corp[43]) as detectors in this version. The comparison between the responses of OPT-101 and ODA-6WB-500M photodetectors are shown in Appendix J. The LEDs remained the same between this and the prior version. Regarding participant comfort, we used single optodes instead of putting them in a circuit in this version. To make the optodes tight, we developed 3D printed holders for each LED and

detector. Figure 22 shows the sensor placement on head cap with the holders and the LED with its holder.

Figure 22 Sensor Placement on the Head Cap and LED holder

3.3.2 Control Unit 1: LED Driver Circuit

We used the same LED driver design configuration as the second version of our systems. However, since the PCB which holds the component shrank, the LED Driver Circuit also shrank. While the dimensions of Version 2 were 10.4 cm x 3.8 cm, Version 3 measured 3.6 cm x 3.7 cm. We significantly reduced the circuit which helped us to develop a smaller portable unit. The programming algorithm to drive and trigger the LEDs through DEMUX and current sources remained the same because we used the exact same configuration. The difference between two circuits can be seen on Figure 23.

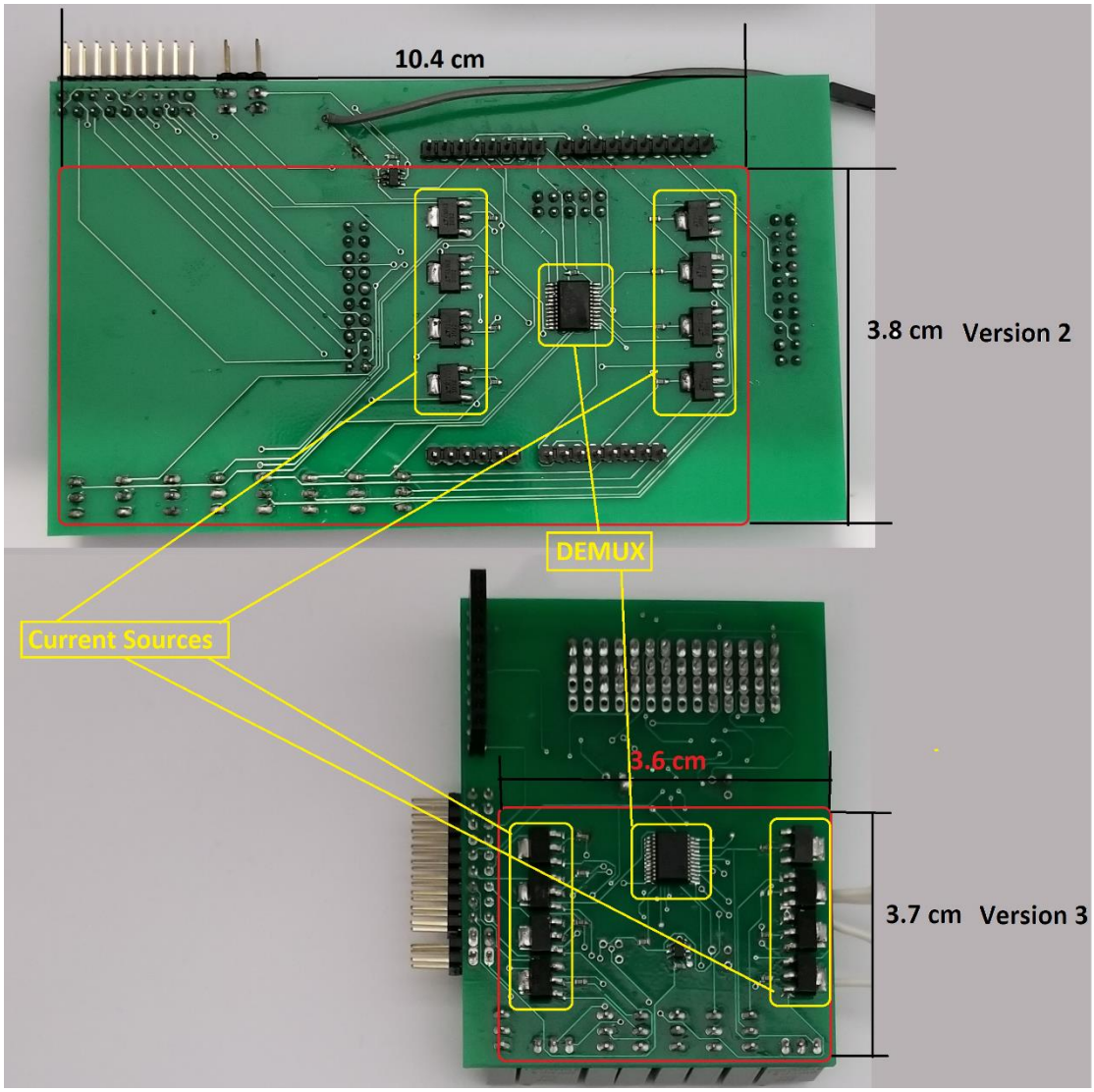


Figure 23 The difference of LED Drive Circuits in Version 2 and Version 3

3.3.3 Control Unit 2: Analog to Digital Converter

The analog to digital converter mentioned in Version 2 uses the evaluation board to connect with the LED driver circuit and the Intel Edison. This evaluation board includes all the necessary supporting components for reliable measurements. It is very useful module, but its size remains a limitation for use in our desired system. Using only the ADC chip is also not convenient because it requires a dedicated time to learn about the necessary connections to implement the same sensitivity with the evaluation board.

Therefore, we decided to use ADS1115 Analog to Digital Converter to minimize the system. The features of ADS1115 are:

- 16 bits
- Programmable data rate: 8 SPS to 860 SPS
- Internal Oscillator
- Four single-ended or two differential inputs

ADS1115 communicates with Intel® Edison via I^2C. Since I^2C communication requires only two wires, we reduced the number of wires [44]. Figure 24 shows the board of ADS1115 designed by Adafruit.

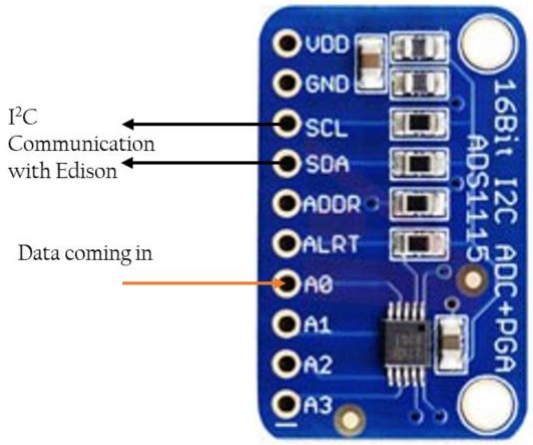


Figure 24 ADS1115 board designed by Adafruit

3.3.4 Control Unit 3: Intel® Edison Compute Module and Breakout Board

A microcontroller in the control unit is used to control the entire system. We used the Intel Edison Compute Module as the microcontroller for our control unit. A daughter board then connected the other components to the Edison. Because the Arduino board is large and has unnecessary components for our project, we chose another daughter board. There are two other options as daughter board for Intel Edison

Compute Module. They are the Intel Mini Breakout Board and the Sparkfun Intel Edison Blocks. The Sparkfun Intel Edison Blocks are smaller than Intel Mini Breakout Board, but they require different boards for every operation like powering the board, making a serial communication with the computer, connecting to other components via GPIO pins etc. This requirement for several blocks increases the size and complexity of the system. For this reason, chose to use the Mini Breakout Board. Figure 25 shows the relative sizes of the Arduino and breakout boards.

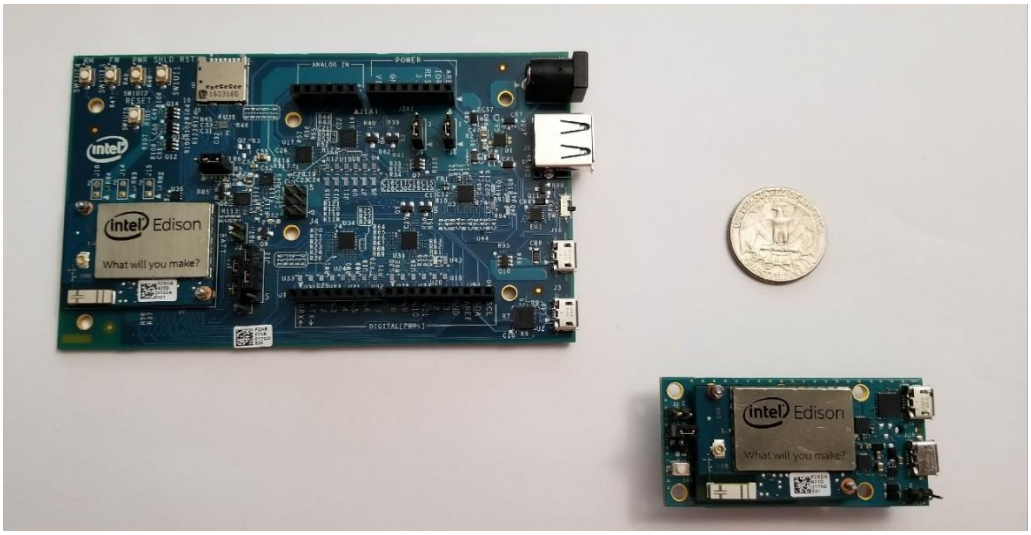


Figure 25 Size comparison of the Intel Edison Arduino and Mini Breakout Board

The main difference between Intel Mini Breakout Board and Intel Arduino Board is their voltage outputs. While the Intel Arduino Board supplies 5V, Intel Mini Breakout Board gives 1.8V. As we experienced in Version 1, this does not trigger a strong enough LED activation. Other components, such as current sources, DEMUX, and the ADS1115 also require at least 3.3V to operate. However, I kept using the Intel Edison Compute Module and its Mini Breakout Board for their convenience in remote programming. The module can be programmed without a physical computer connection, allowing programming to take place without interrupting experiments. In

order to generate enough power from the Intel Mini Breakout Board, we used voltage level shifters to convert 1.8V to 5V. We chose to use a TXB0108 8-bit, bi-directional voltage level shifter (Texas Instruments) with the DEMUX and a TXS0104 4-bit, bi-directional voltage level shifter (Texas Instruments) with the ADS1115. We used two different translators because TXB0108 is not compatible with I^2C communication and we needed at least a 5-bit translator for the DEMUX. Verification tests of Intel Edison Breakout Board with LED driver circuit and ADS1115 are described in Appendix G.

We combined the three boards (LED Driver Board, Analog-to-Digital Converter and Edison Breakout Board). Figure 26 shows the sandwiched system.

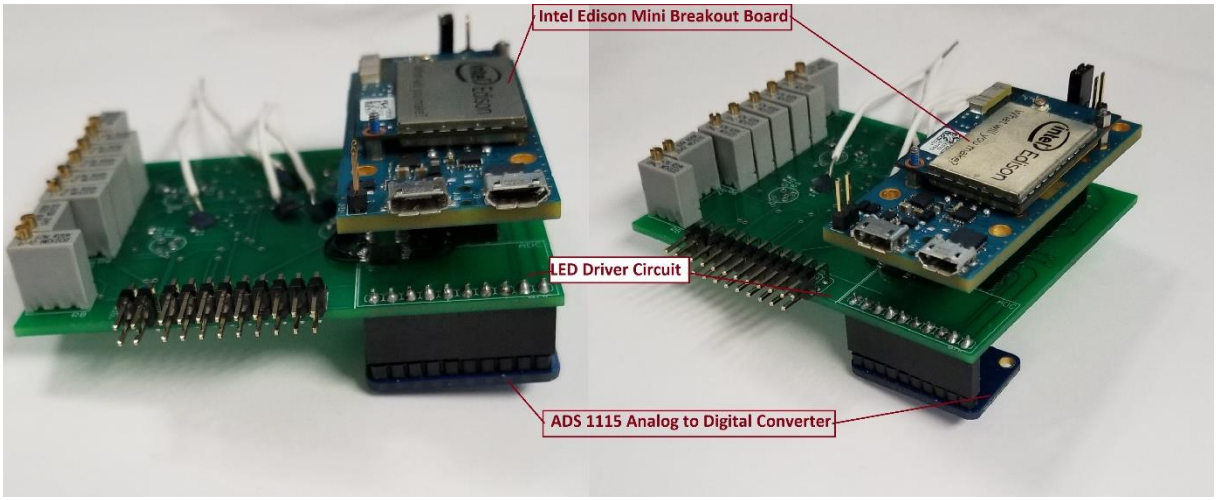


Figure 26 Sandwiched System of Version 3

The schematic and layouts are given in Appendices H.

3.3.5 Computing Unit: Software

Because the Intel Mini Breakout Board is connected to the Intel Edison Compute Module, it is programmable with the Arduino IDE. The only difference between programming the Intel Mini Breakout Board and the Intel Arduino Board is their header

pin numbers. A detailed pin number comparison table is provided in Appendix I. The flow chart of the code for collecting raw data is provided in Figure 27.

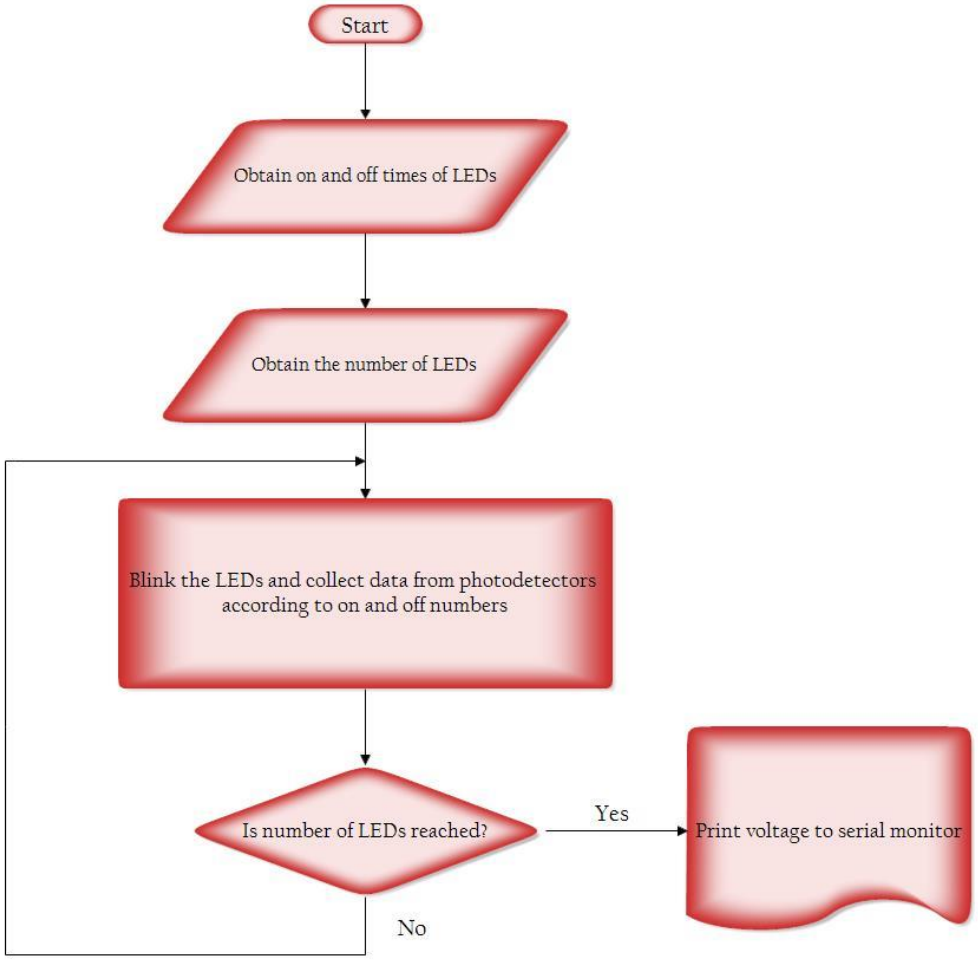


Figure 27 Flowchart of raw data collection

We also tested Edison Breakout Board for wireless programming. It works with a 9V rechargeable battery and can be programmed via command line programs such as Terminal (for Linux and Mac) or Putty (Windows) without using any IDE. The Intel Edison Compute Module has Yocto, an open-source collaboration project to create Linux-based systems for embedded hardware systems [45]. Therefore, to reach the shell of the Intel Edison Compute Module, we used Linux commands. Inside the shell, several programming languages such as C, C++, Python, Node.js and Java can program Edison.

A Linux library, MRAA [46], was used programming with these languages. We chose Python because it is open and easy to use.

Figure 28 Intel® Edison Breakout Board and 9V Rechargeable Battery

CHAPTER 4

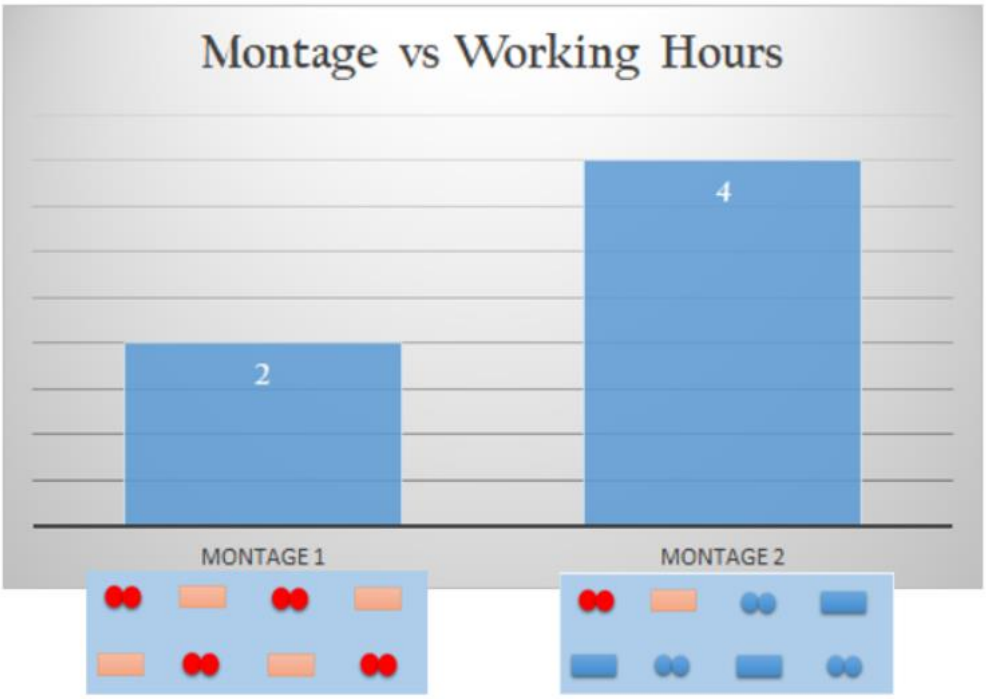
FINDINGS

4.1 Experiments on fNIRS Embedded Computing System

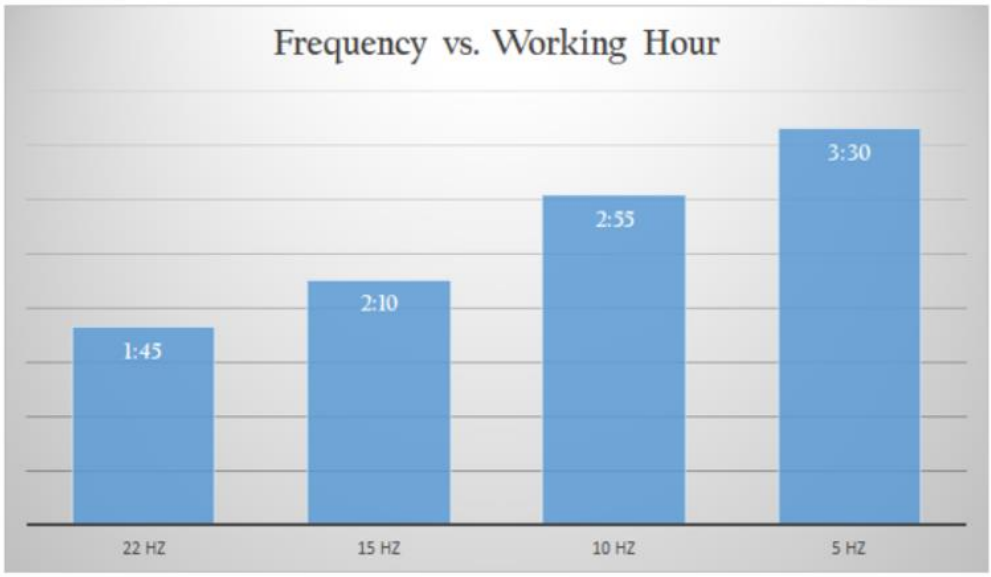
4.1.1 Battery Experiments

Portable systems should work with rechargeable batteries, allowing use when walking, jogging, or cycling. It should not have to be plugged into a wall outlet during use.

To validate the battery life, we did the battery experiments with different montages and sampling rates. In the first montage, all four LEDs and all four detectors were used, blinking in sequence as previously described. In the second montage, only one LED-photodetector pair was used. Both wavelengths were still used in the montage. For overall sampling rates were tested in the single pair montage:22 Hz, 15 Hz, 10 Hz and 5 Hz. Figure 29 shows the results.



(a)



(b)

Figure 29 Battery testing with (a) different montages and (b) different frequencies

4.1.2 System Testing

We verified the functionality of the data collection system before designing the PCB. To do experiments, the ADS1115 analog to digital converter and the DEMUX CD74HC4067 were put together on the breadboard. The TXB0108 and TXS0104 were added for voltage level translation. The Intel Edison Mini Breakout Board was then connected to the breadboard and components via jumper wires. Figure 30 shows the testing design.

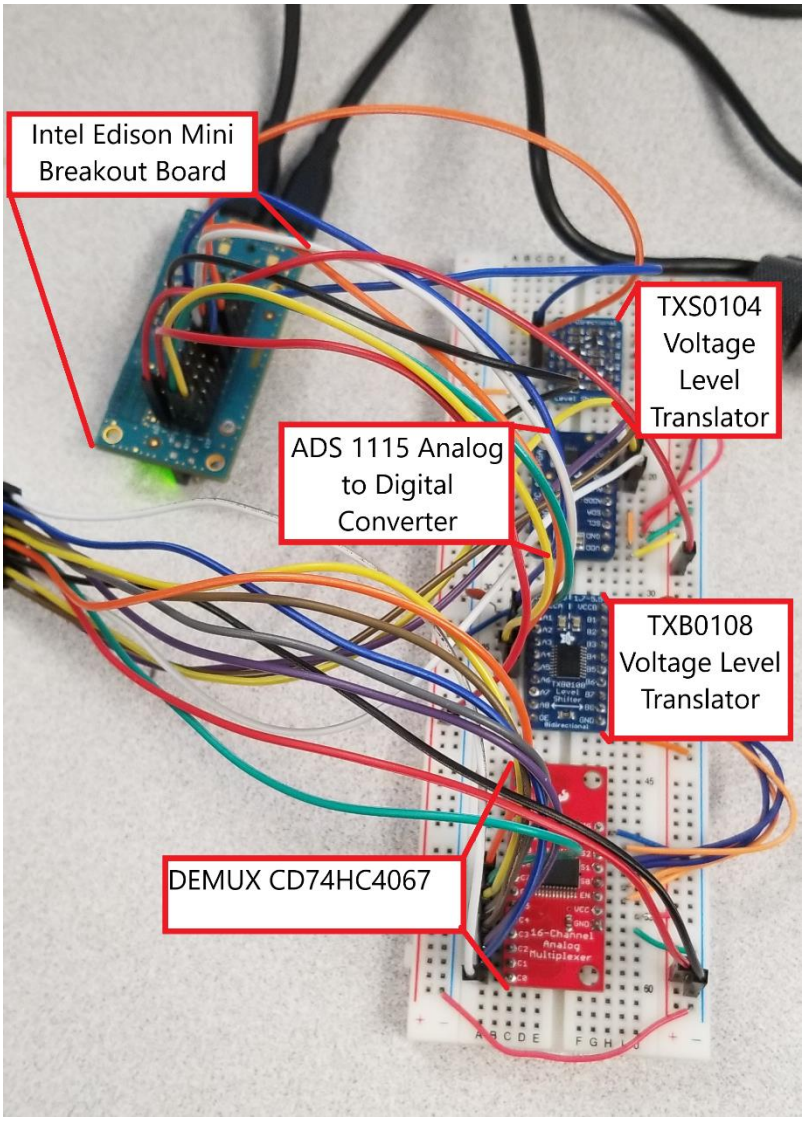
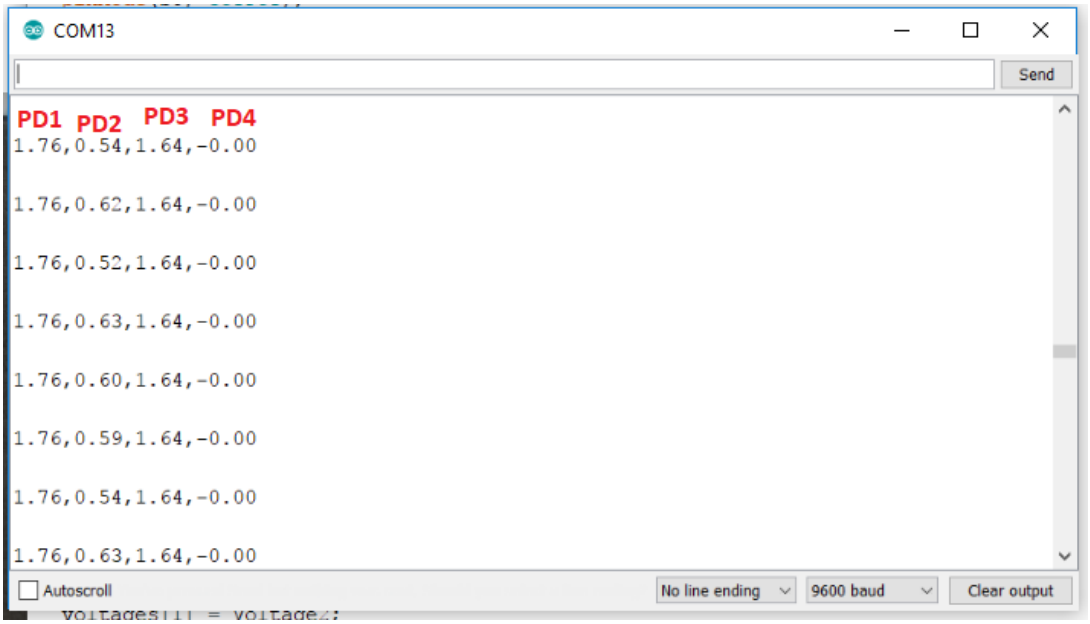


Figure 30 Testing the system on the breadboard

We verified that the Intel Edison was sending the signal to the DEMUX to set the switches. The DEMUX was setting the channels according to source configuration and sending the signal to blink the LEDs. The data on the serial monitor verified that the photodetectors detected the light coming from the LEDs and sent them to ADS1115 analog to digital converter. It also verified that the ADC digitized the signals and sent them to the Intel Edison Breakout Board via I^2C communication. We then measured the voltage level of the photodetector output, ADC output and Intel Edison input pins and verified that it showed the same value with the serial monitor. Figure 31 shows the blinking headband and photodetector data.

(a)



(b)

Figure 31 (a) Blinking LEDs on the headband, (b) the values from the photodetectors

4.1.2.a Algorithmic Configurability

We programmed the system for different configurations.

4.1.2.a.i Number of Channels

In this study, we used 4 multi-wavelength LEDs which have 2 wavelengths, so we had 8 output channels in total. Each LED output channel was connected to a channel of the DEMUX. Thus, we were able to control all wavelengths with code digitally. The number of LEDs was determined in the code with the “pinCount” variable and put in a for loop function. Figure 32 shows the screenshot of the specific part of the code. (Full code can be found in Appendix C)

```
int E = 7;

int i = 25;
int j = 5;
int pinCount;
```

```
for (pinCount = 0; pinCount < 8; pinCount++){
  firstsource(); //LED1 is ON
  delay(i);
  adc0 = ads.readADC_SingleEnded(0);
  Voltage1 = (adc0 * 0.1875)/1000;
  adc1 = ads.readADC_SingleEnded(1);
  Voltage2 = (adc1 * 0.1875)/1000;
  adc2 = ads.readADC_SingleEnded(2);
  Voltage3 = (adc2 * 0.1875)/1000;
  adc3 = ads.readADC_SingleEnded(3);
  Voltage4 = (adc3 * 0.1875)/1000;
  sendDataToSerial('A', Voltage1);
  sendDataToSerial('B', Voltage2);
  sendDataToSerial('C', Voltage3);
  sendDataToSerial('D', Voltage4);
```

Figure 32 Piece of code which determines the number of channels

We also used 4 photodetectors. Therefore, for each wavelength, there were 4 photodetector values, for a 32 data channels. We separated the detector values by labelling them with letters “A... D”. Therefore, by editing these numbers, we can change the configuration. Figure 33 shows the display of different detectors.

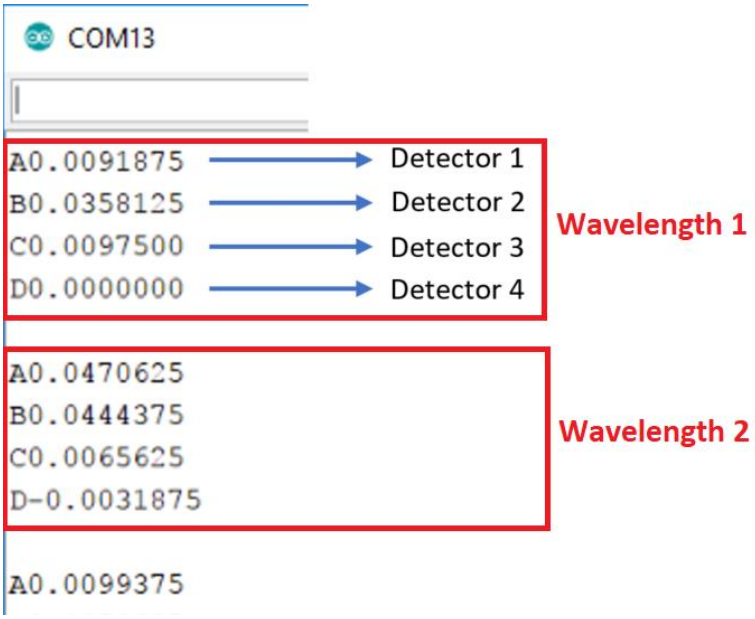


Figure 33 Displaying of different detectors

4.1.2.a.ii Sampling Rate

The sampling rate depends on the LEDs on and off times. These sampling rate can therefore be controlled by configuring the on and off time variables in the code. In the code, we defined “i” and “j” variables as on time and off times, respectively. Then we put them in “delay” function to tell the system to wait during these times. The unit of delay function in the Arduino IDE is milliseconds. In this study, we used a 20ms on time and 5ms off time, per LED and per wavelength. Therefore, the total timing is 200ms, and the overall frequency is 5 Hz. Figure 34 shows the screenshot of that piece of the code.

```
int i = 25;
int j = 5;
int pinCount;
```

```
firstsource(); //LED1 is ON
delay(i);
adc0 = ads.readADC_SingleEnded(0);
Voltage1 = (adc0 * 0.1875)/1000;
voltages[0] = Voltage1;
adc1 = ads.readADC_SingleEnded(1);
Voltage2 = (adc1 * 0.1875)/1000;
voltages[1] = Voltage2;
adc2 = ads.readADC_SingleEnded(2);
Voltage3 = (adc2 * 0.1875)/1000;
voltages[2] = Voltage3;
adc3 = ads.readADC_SingleEnded(3);
Voltage4 = (adc3 * 0.1875)/1000;
voltages[3] = Voltage4;
Serial.print(voltages[0]);
Serial.print(",");
Serial.print(voltages[1]);
Serial.print(',');
Serial.print(voltages[2]);
Serial.print(",");
Serial.println(voltages[3]);
Serial.println();
nochannel(); //LED1 is OFF
delay(j);
secondsource(); //LED2 is ON
delay(i);
```

Figure 34 Code determining the sampling rate

4.2. Experiments on Wireless Communication

The wireless communication between computer and Intel Edison Compute Module was established via Putty using the board's IP address. The command line on Putty and SSH connection is shown in Figure 35.

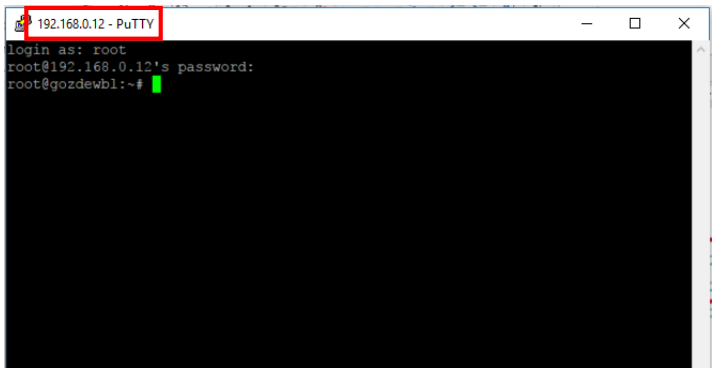


Figure 35 SSH connection login screen

The Intel Edison Compute Module was programmed with Python to trigger the LEDs, collect the data and show on the screen. The command to run the program and output of the program is demonstrated in Figure 36.

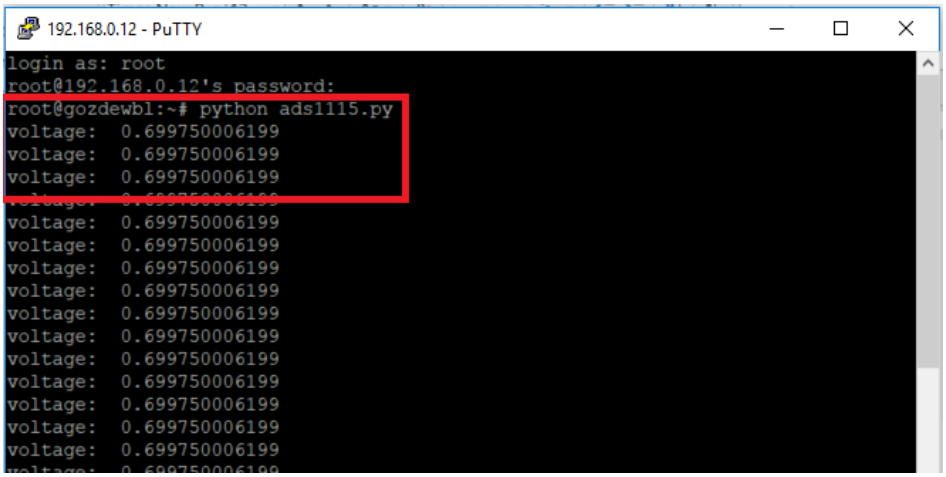


Figure 36 Output of Python Code

The Intel Edison Compute Module also could communicate with computer via TCP/IP communication. I was able to display the output and store it in a text file successfully. Figure 37 shows the command prompt, Putty windows and text file together.

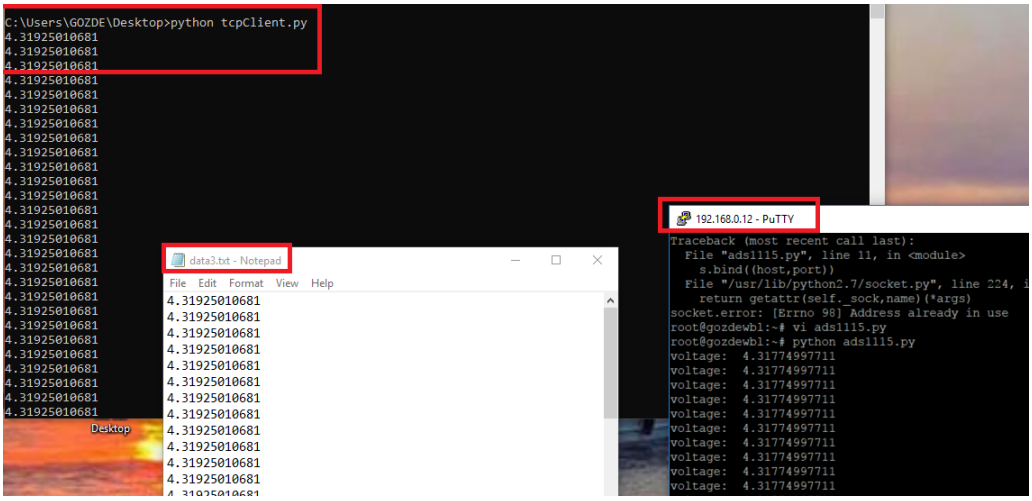


Figure 37 Command Prompt, Putty and Text File

To graph the data, a Matlab custom script was used. The data coming from fNIRS control unit was received by Matlab via TCP/IP communication. Then Matlab drew the graph. Figure 38 demonstrates the data and graph.

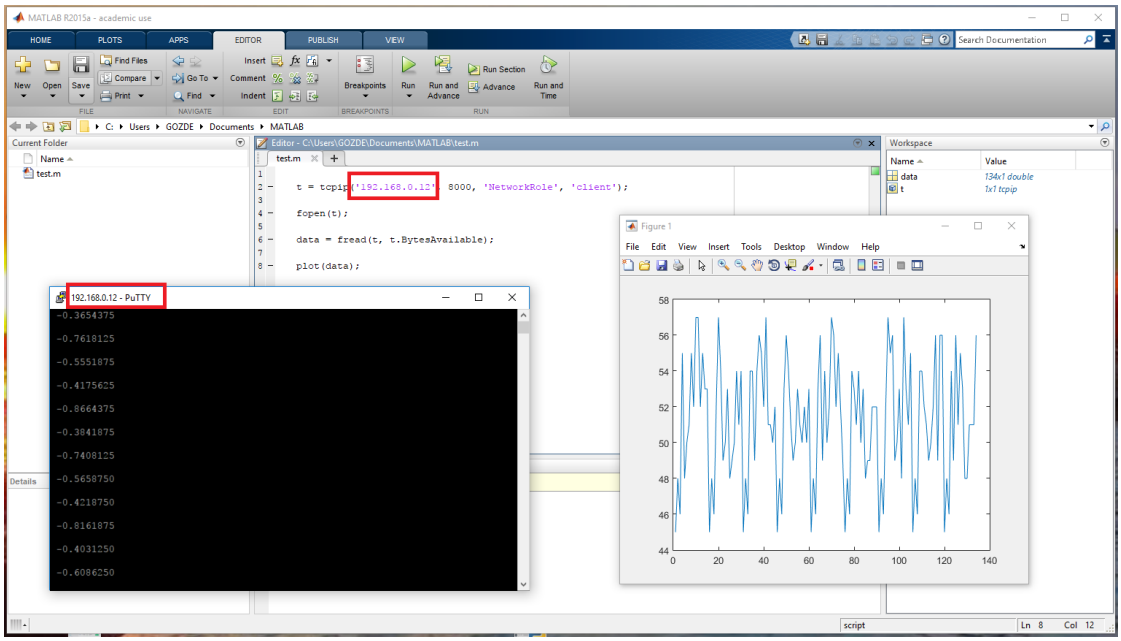


Figure 38 Wireless Communication and Data Graph

4.3. Feasibility Study on Human Subjects

Pulse rate and arterial occlusion experiments were done to validate our system's feasibility on human subjects.

4.2.1 Pulse Rate Experiments on Finger

In this experiment, the fNIRS sensor pad was placed on the finger to detect oxygenation and pulse rate. The system was able to detect pulse signal successfully. The signal is demonstrated in Figure 39.

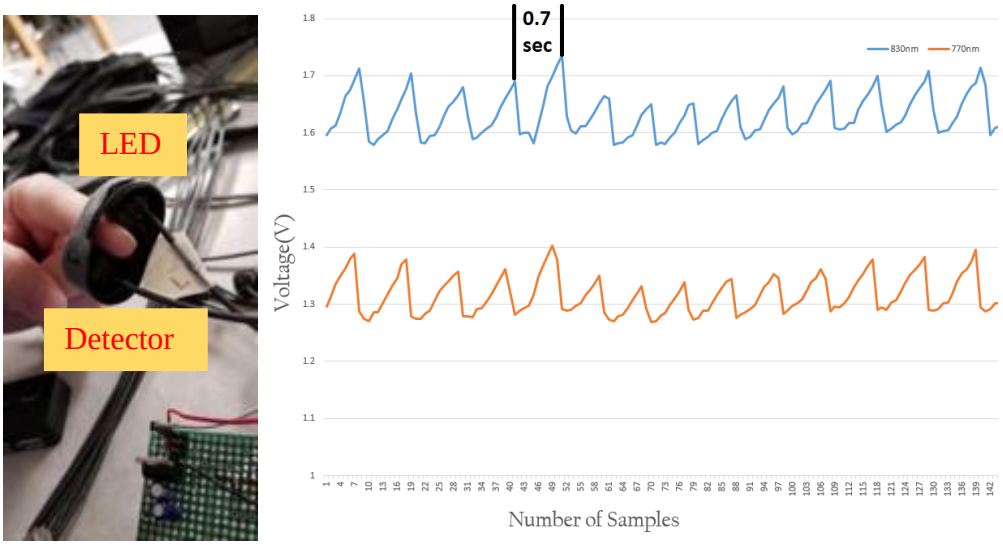


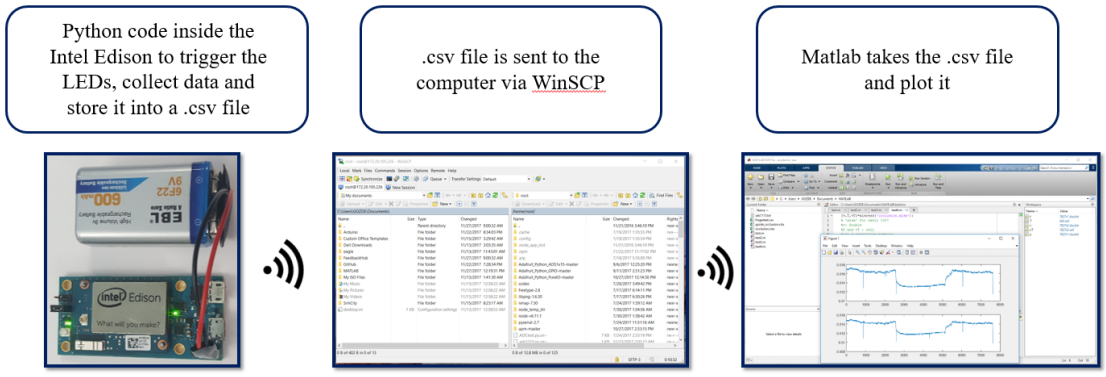
Figure 39 Pulse signal obtained from fNIRS sensor

To calculate heart rate, the time difference between two pulses was obtained. There was 0.7 second between two pulses. The equation to calculate heart rate is given below.

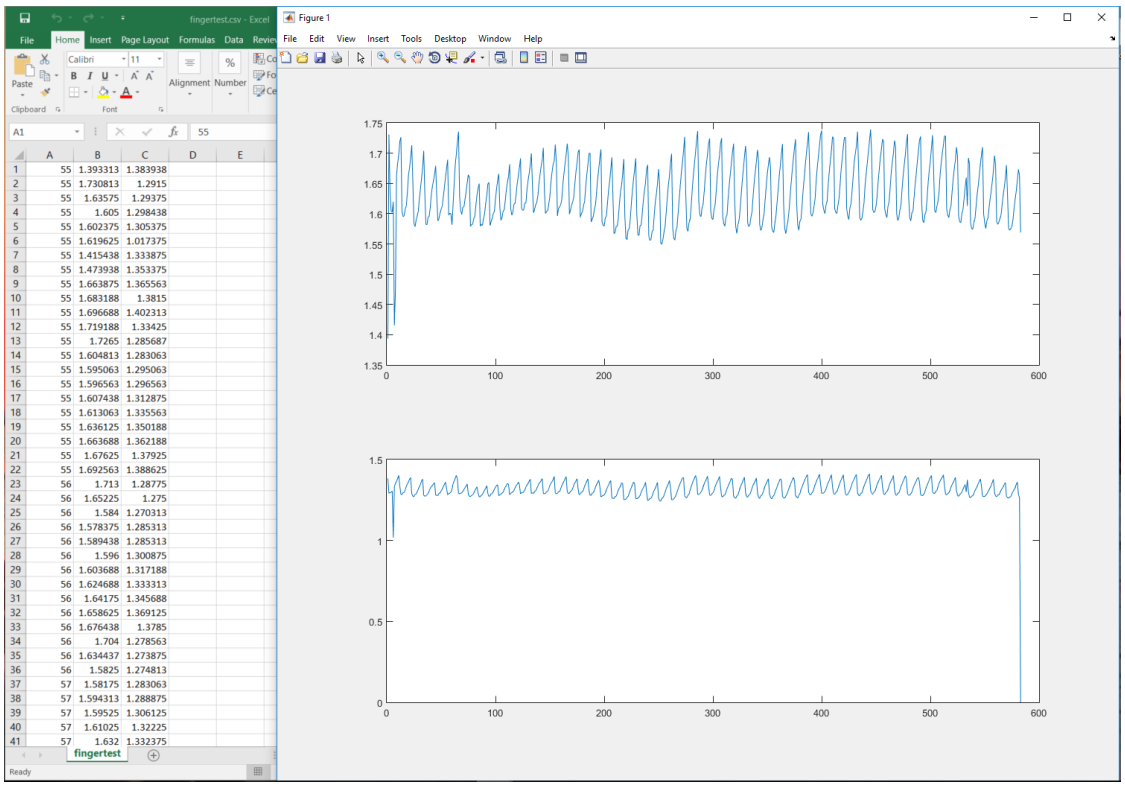
$$HR = (1/RR\ (sec)) \times 60\ BPM \quad (4)$$

According to equation 4, heart rate was calculated as 84.5 BPM. The resting heart rate for a healthy person is between 60 and 100 BPM, so the value is in the appropriate range [47].

For IoT fNIRS, the experiment was performed wirelessly. The Intel Edison triggered the LEDs, collected the data from the ADC, displayed them on the screen and stored them into a .csv file. Then the .csv file was copied to computer via WinSCP. Then, the copied .csv file was read by Matlab and graphed. Figure 40 shows the block diagram of wireless communication, the .csv file and Matlab figure.



(a)



(b)

Figure 40 (a) Block Diagram of Wireless Communication (b) The .csv file and Matlab figure

4.2.2 Experiments for Arterial Occlusion

In medicine, occlusion means that the blockage or closing of a blood vessel or hollow organ [48]. For arm, when the artery is blocked, the blood flow going through the fingers is getting slower depends on the amount of pressure. Therefore, while the

amount of oxyhemoglobin decreases, the amount of deoxyhemoglobin increases during the occlusion. When the artery is released, the blood flow rushes, the amount of oxyhemoglobin increases, and the amount of deoxyhemoglobin decreases. It is possible to catch the changes in oxy and deoxyhemoglobin with fNIRS system.

In this experiment, the fNIRS sensor was placed on the subject's arm, below the elbow. Also, a blood pressure monitor was placed above the elbow on the muscle to occlude the artery during the experiment. To capture the oxygenation changes during occlusion, a 9-minutes experiment protocol was applied. We experimented while the subject was sitting in a relaxed position. The trial was started with 3 min rest block. After 3 minutes, occlusion was applied for another 3 min. During the occlusion, subject stayed still. This was followed by 3 min rest block. Therefore, one trial lasted for 9 minutes.

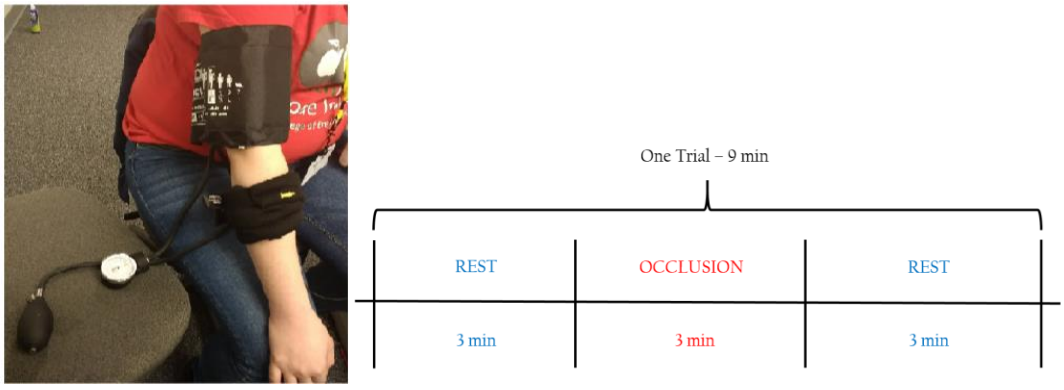


Figure 41 Experiment setup and experimental protocol for arterial occlusion

For IoT fNIRS, this experiment was also performed wirelessly. The Intel Edison triggered the LEDs, collected the data from the ADC, displayed them on the screen and stored them into a .csv file. Then the .csv file was copied to computer via WinSCP. Then, the copied .csv file was read by Matlab and graphed. Figure 42 shows the csv file and Matlab figure with raw data.

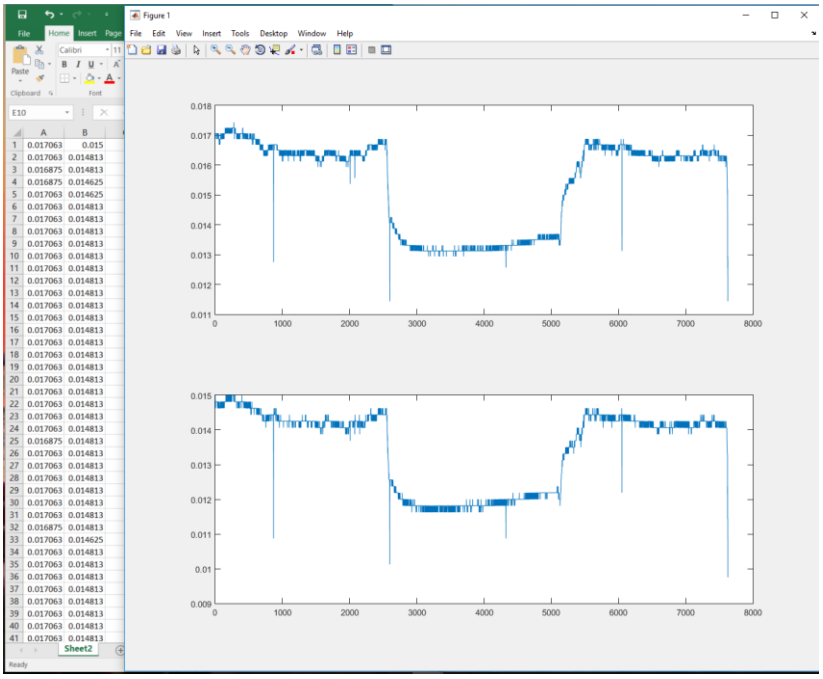
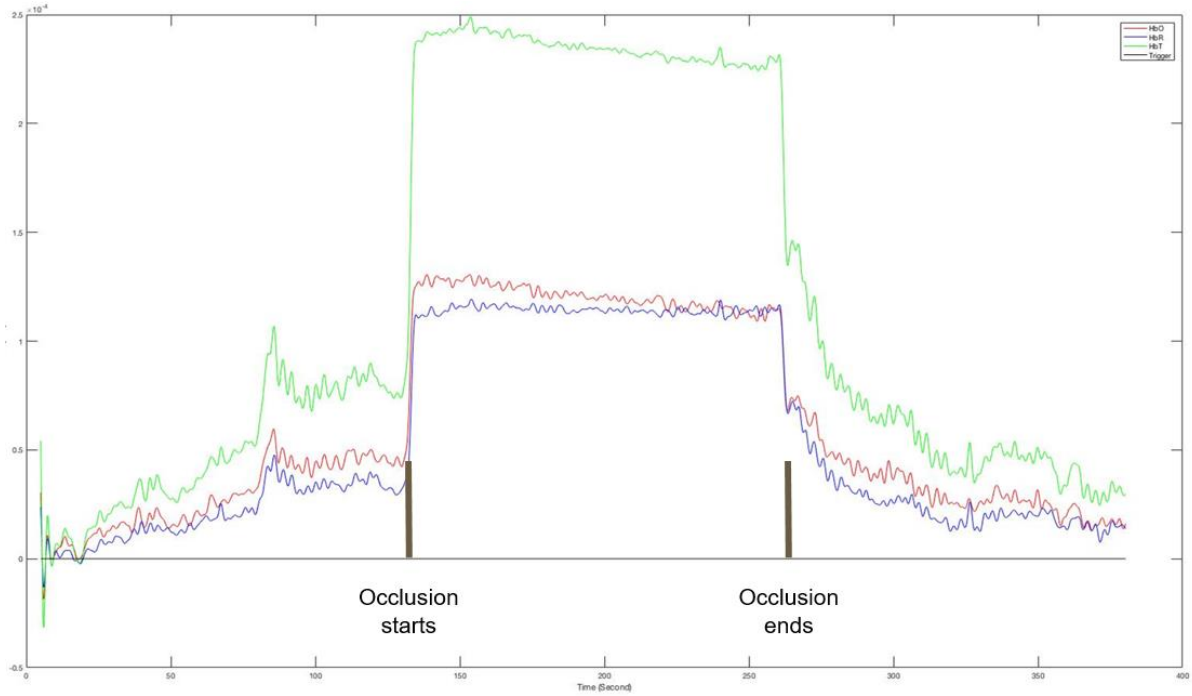
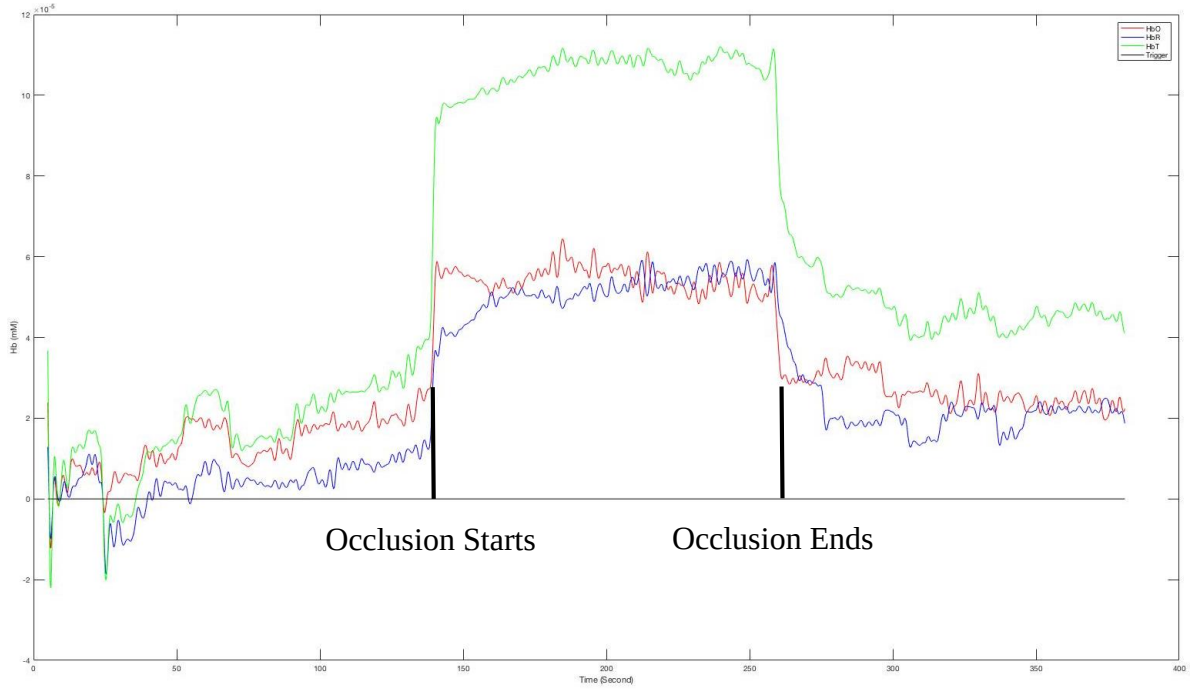


Figure 42 The .csv file and Matlab figure with raw data

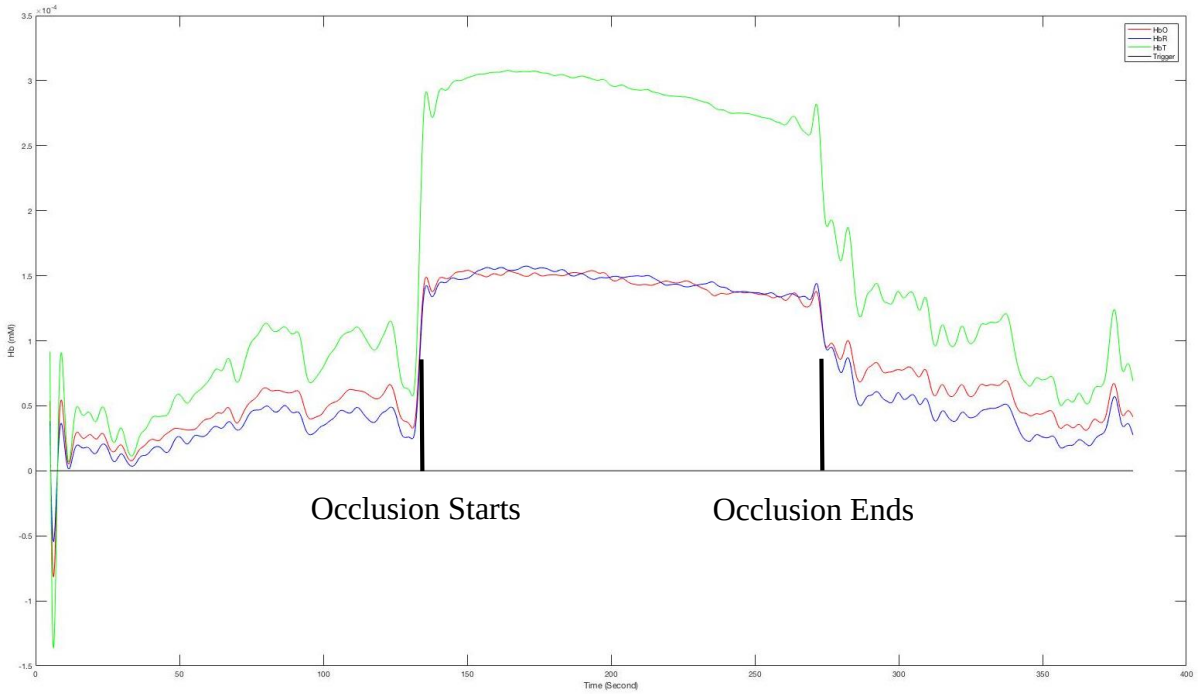
The system was able to detect the decrease of oxygenation during the occlusion and increase of oxygenation when the artery was released successfully. This result is in agreement with previous arterial occlusion experiments [49]. The hemodynamic changes from different subjects are illustrated in Figure 43.


HbO
HbR
HbT
Trigger
Occlusion
starts
Occlusion
ends
Time (Second)

(a)


HbO
HbR
HbT
Trigger
Hb (mM)
Occlusion Starts
Occlusion Ends
Time (Second)

(b)



(c)

Figure 43 (a) Subject # 1 (b) Subject # 2 (c) Subject # 3

CHAPTER 5

CONCLUSION

Functional near infrared spectroscopy (fNIRS) is a noninvasive functional brain monitoring method to monitor brain's hemodynamic changes. Due to the fact that fNIRS has relatively high spatial and temporal resolution and it is portable, it is used as a successful alternative of fMRI and EEG to cover their disadvantages. However, current fNIRS devices require long cables to connect the fNIRS sensor pad to control unit which is used for controlling the sensor and collecting the data, so it is limited to do experiments in an ambulatory environment such as walking, jogging and cycling. Moreover, because they are tabletop devices, patients or participants need to go to the hospital to participate the experiments. To address these limitations, we developed a small, wearable, battery-operated fNIRS system which can be used in mobile experiments.

The devised system is comprised of three parts: an fNIRS sensor pad, a control unit and a computing unit. A flexible foam headband which carries the optodes (LEDs and photodetectors) is designed as fNIRS sensor. The control unit is responsible for controlling the commands to send signal to LEDs and collect the data from photodetectors. The sensor connects to a control unit via wires. The control unit contains three main units. To increase the light intensity emitted to the brain, an LED driver circuit is designed. To convert the analog data coming from photodetectors to digital data which the microcontroller can perform further processing on, ADC is used. To control all commands and make communications between control unit and computing

unit, a microcontroller is used. The control unit connects to computing unit via serial communication. The computing unit includes the software which is used for data acquisition and data display.

To validate the implemented system, experiments on system level and data verification were performed. On the system level, we tested the battery life and algorithmic configurability. According to the test results, it was proven that the system could work up to 4 hours on battery. Also, the system was found to perform with different configurations. We were able to adjust the number of channels and sampling rate. For data verification, we performed finger pulse rate and arterial occlusion tests. The system was able to detect the pulses from the finger successfully. As the result of arterial occlusion test, the system detected a sudden decrease of oxygen level due to arterial occlusion and sudden increase of oxygen level due to the release of the artery. In conclusion, the system answers the portable fNIRS system requirements which was mentioned in Chapter 3.

5.1 Research Insights

The original purpose of this study was to create a small, wireless, wearable fNIRS measurement system and validate it. However, it was seen that in order to make accurate and reliable measurements, we need powerful microcontrollers.

The major drawback that we faced during this study is the output voltage of Intel Edison Breakout Board. The output voltage value is 1.8V, and most of the components require 3.3V or 5V to operate. For this reason, we had to use voltage level translators, which made our circuit bigger. Also, it requires extra configuration and programming. We tried to use transistors as level shifters first, but the direction of the current level was

not appropriate for our system. Figure 44 shows the working principle of a PNP transistor.

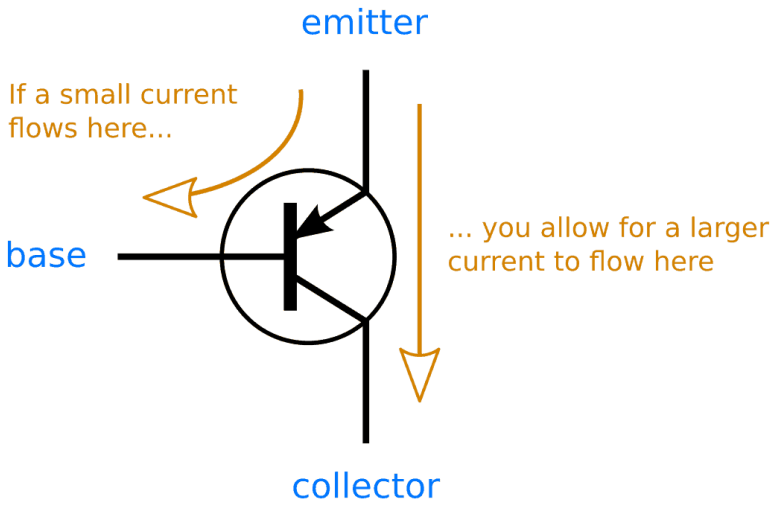


Figure 44 Working principle of a PNP transistor

As can be seen from the figure, the current should flow from emitter to base and out from the base in order to work the transistor [50]. However, in our system, the current should flow into the base from Edison. It is the working principle of an NPN transistor, but in NPN transistor, emitter must be connected to ground in order to supply current flow [51]. Figure 45 shows the working principle of an NPN transistor.

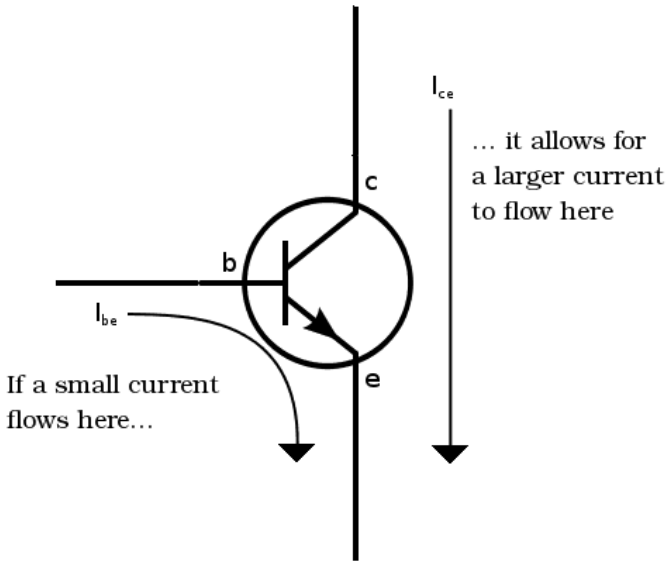


Figure 45 Working principle of an NPN transistor

In our system, we need to connect emitter to VCC because of our current flow. For these reasons, it is impossible to use a PNP or NPN transistor as voltage level shifter.

Therefore, we switched to voltage level shifter ICs for level shifting. We found TXB0108 Bidirectional Voltage-Level Shifter from Texas Instrument. Its 8-bit is so ideal for our system since we need 2 bits for ADS1115 and 5 bits for DEMUX pins. However, after testing with ADS1115, we realized that TXB0108 is not compatible with I^2C communication. For this reason, we needed to search about another voltage level shifter IC which is compatible with I^2C communication. We found that TXS0104 is appropriate for I^2C communication; however, it has only four bits so not enough for our system. Thus, we used TXB0108 for DEMUX and TXS0104 for ADS1115.

In design of our PCB, we designed the components according to their footprints in Eagle. Also, we ordered them according to their footprints. For TXS0104, the footprint in the Eagle was named wrong. They named it as TSSOP instead of SOIC. So the actual component was smaller than footprint. We had to find a quick solution with jumper wires. Figure 46 shows the footprint and actual size of TXS0104 chip.

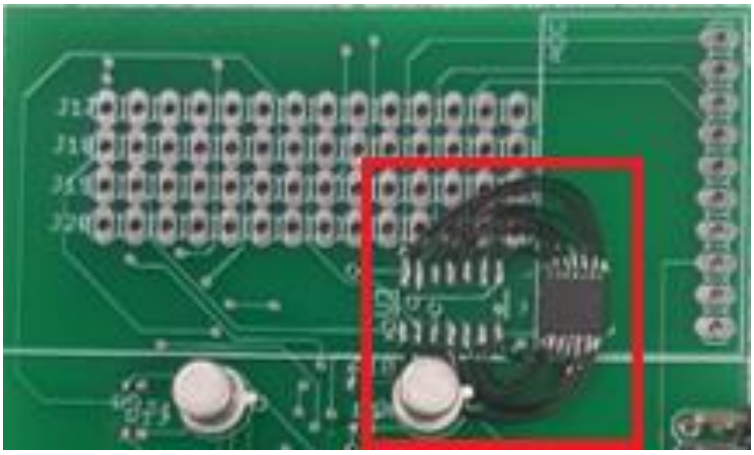


Figure 46 The footprint and actual size of TXS0104

The jumper wires on the PCB not only make it harder to run but also look incomplete and amateur on the PCB. So, the footprints in the Eagle should be reviewed very carefully.

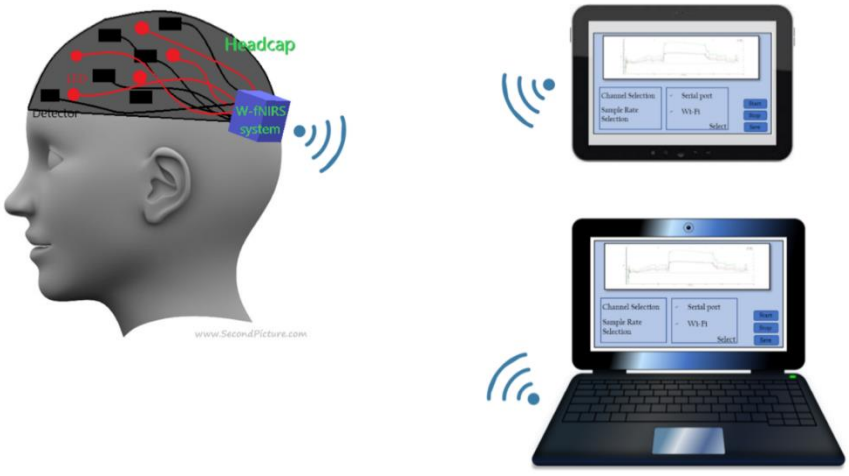
Another problem is about the wireless communication. To supply wireless communication and wireless programming, we decided to use MRAA library [46]. MRAA library allows us to program Intel Edison from the terminal without any IDE. However, the Python library for ADS1115 and I^2C communication is not compatible with MRAA. So we were not able to program Intel Edison with MRAA library. Due to the fact that Intel Edison discontinued, there is no development in libraries for Intel Edison [52]. Also, because Intel Edison Breakout Board uses the same cable for both powering and serial communication, it is impossible to reach serial monitor when the board is powered with an external battery. This situation makes the battery experiments harder. For these reasons, changing the microprocessor and using ARM processors should be considered seriously.

5.2 Future Work

The achievements of current work and suggestions for future work are mentioned in Table 4 and Figure 47.

Table 4 The achievements of current work and suggestions for future work

Areas	Current Work	Future Work
Hardware	* A smaller-size-of PCB was designed to shrink all necessary components to control fNIRS system. It increased the portability. * The microprocessor which controls the fNIRS system was improved through the evaluation of three systems. So we were able to calculate accurate and reliable data	* An extremely small PCB which can be placed into the head cap will be designed. Thus, we will reduce the length of cables between the control unit and fNIRS sensor. * Better holder design and circuit design for fNIRS sensor, LEDs and detectors, will be created.
Software	* An embedded software was developed for fNIRS firmware so we could configure most features by ourselves. Gave us freedom to configure the measurements	* A Graphical User Interface (GUI) will be developed in Matlab or Python to display and track real-time data. Also, it will be configurable from the GUI. * To support portability, real-time wireless communication between fNIRS system and computer/smartphone will be established
Design	* A portable fNIRS system was designed, so the ease of use in the natural environment was increased.	* The fNIRS system will be wireless and thus more portable. Also, it will be more configurable by the end-users.



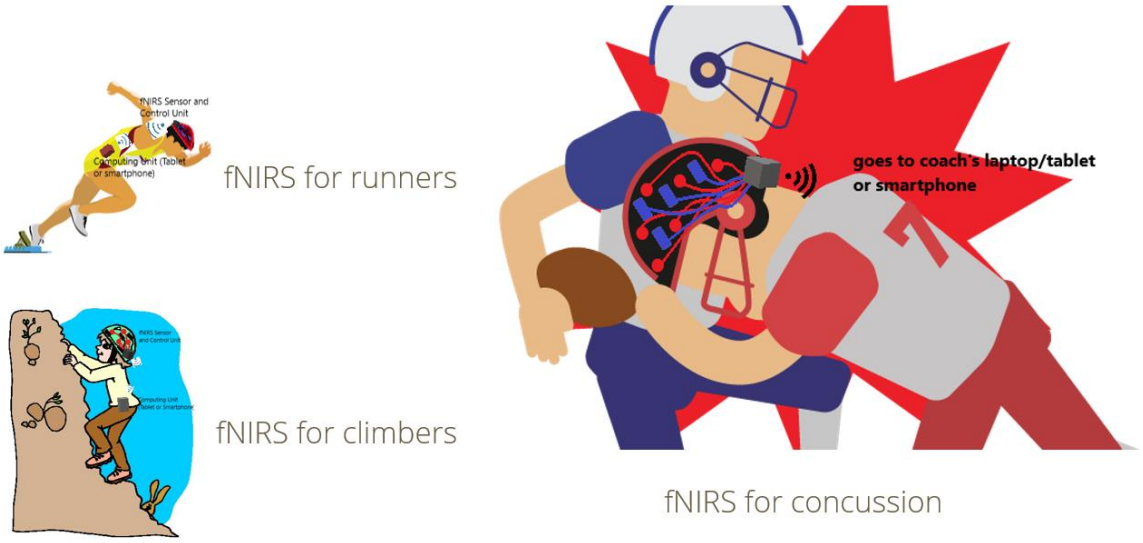


Figure 47 Future of W-fNIRS system

Appendix A: **DEMUX Testing Code**

```
int s0 = 3;
int s1 = 5;
int s2 = 6;
int s3 = 9;
int E = 10;

void firstsource(){
 digitalWrite(E, LOW);
 digitalWrite(s0, HIGH);
 digitalWrite(s1, LOW);
 digitalWrite(s2, LOW);
 digitalWrite(s3, LOW);
 Serial.println("out1");
}

void secondsource(){
 digitalWrite(E, LOW);
 digitalWrite(s0, LOW);
 digitalWrite(s1, HIGH);
 digitalWrite(s2, LOW);
 digitalWrite(s3, LOW);
 Serial.println("out2");
}

void thirdsource(){
 digitalWrite(E, LOW);
 digitalWrite(s0, LOW);
 digitalWrite(s1, LOW);
 digitalWrite(s2, HIGH);
 digitalWrite(s3, LOW);
 Serial.println("out3");
}

void fourthsource(){
 digitalWrite(E, LOW);
 digitalWrite(s0, HIGH);
 digitalWrite(s1, LOW);
 digitalWrite(s2, HIGH);
 digitalWrite(s3, LOW);
 Serial.println("out4");
}
```

```
void fifthsource(){
 digitalWrite(E, LOW);
 digitalWrite(s0, LOW);
 digitalWrite(s1, HIGH);
 digitalWrite(s2, HIGH);
 digitalWrite(s3, LOW);
 Serial.println("out5");
}

void sixthsource(){
 digitalWrite(E, LOW);
 digitalWrite(s0, LOW);
 digitalWrite(s1, LOW);
 digitalWrite(s2, LOW);
 digitalWrite(s3, LOW);
 Serial.println("out6");
}

void seventhsource(){
 digitalWrite(E, LOW);
 digitalWrite(s0, HIGH);
 digitalWrite(s1, HIGH);
 digitalWrite(s2, LOW);
 digitalWrite(s3, LOW);
 Serial.println("out7");
}

void eighthsource(){
 digitalWrite(E, LOW);
 digitalWrite(s0, HIGH);
 digitalWrite(s1, HIGH);
 digitalWrite(s2, HIGH);
 digitalWrite(s3, LOW);
 Serial.println("out8");
}

void nochannel(){
 digitalWrite(E, HIGH);
 Serial.println("no channel");
}

int pinCount;

void setup() {
 pinMode(s0, OUTPUT);
```

```
 pinMode(s1, OUTPUT);
 pinMode(s2, OUTPUT);
 pinMode(s3, OUTPUT);
 pinMode(E, OUTPUT);

 Serial.begin(9600);
}

void loop() {

for (pinCount = 0; pinCount < 8; pinCount++) {
 firstsource(); //first LED on
 delay (250);
 nochannel(); //first LED off
 delay (250);
 secondsource(); //second LED on
 delay (250);
 nochannel(); //second LED off
 delay (250);
 thirdsource(); //third LED on
 delay (250);
 nochannel();  //third LED off
 delay (250);
 fourthsource(); //fourth LED on
 delay (250);
 nochannel(); //fourth LED off
 delay (250);
 fifthsource(); //fifth LED on
 delay (250);
 nochannel(); //fifth LED off
 delay (250);
 sixthsource(); //sixth LED on
 delay (250);
 nochannel(); //sixth LED off
 delay (250);
 seventhsource(); //seventh LED on
 delay (250);
 nochannel(); //seventh LED off
 delay (250);
 eighthsource(); //eighth LED on
 delay (250);
 nochannel();
 delay (250); //eighth LED off
 }
 }
```

Appendix B: **ADS1115 Testing Code**

```
#include <Wire.h>
#include <Adafruit_ADS1015.h>

Adafruit_ADS1115 ads(0x48);
float Voltage = 0.0;

void setup(void)
{
 Serial.begin(9600);
 ads.begin();
}

void loop(void)
{
 int16_t adc0;  // we read from the ADC, we have a sixteen bit integer as a result

 adc0 = ads.readADC_SingleEnded(0);
 Voltage = (adc0 * 0.1875)/1000;

 Serial.print("AIN0: ");
 Serial.print(adc0);
 Serial.print("\tVoltage: ");
 Serial.println(Voltage, 7);
 Serial.println();

 delay(100);
}
```

Appendix C: **Raw Data Collection Code**

```
#include <Wire.h>
#include <Adafruit_ADS1015.h>

Adafruit_ADS1115 ads(0x48);
float Voltage1 = 0.0;
float Voltage2 = 0.0;
float Voltage3 = 0.0;
float Voltage4 = 0.0;
float voltages[] = {0,0,0,0};
int s0 = 3;
int s1 = 5;
int s2 = 6;
int s3 = 9;
int value = 0;
int E = 10;

int i = 25;
int j = 5;

void setup(void)
{
  Serial.begin(9600);
  ads.begin();
  pinMode(s0, OUTPUT);
  pinMode(s1, OUTPUT);
  pinMode(s2, OUTPUT);
  pinMode(s3, OUTPUT);
  pinMode(E, OUTPUT);
}

void loop(void)
{
  int16_t adc0;  // we read from the ADC, we have a sixteen bit integer as a result
  int16_t adc1;
  int16_t adc2;
  int16_t adc3;

  firstsource(); //LED1 is ON
  delay(i);
  getADCvalue();
  nochannel(); //LED1 is OFF
  delay(j);
  secondsource(); //LED2 is ON
  delay(i);
```

```
  getADCvalue();
  nochannel(); //LED2 is OFF
  delay(j);
  thirdsource(); //LED3 is ON
  delay(i);
  getADCvalue();
  nochannel(); //LED3 is OFF
  delay(j);
  fourthsource(); //LED4 is ON
  delay(i);
  getADCvalue();
  nochannel(); //LED4 is OFF
  delay(j);
  fifthsource(); //LED5 is ON
  delay(i);
  getADCvalue();
  nochannel(); //LED5 is OFF
  delay(j);
  sixthsource(); //LED6 is ON
  delay(i);
  getADCvalue();
  nochannel(); //LED6 is OFF
  delay(j);
  seventhsource(); //LED7 is ON
  delay(i);
  getADCvalue();
  nochannel(); //LED7 is OFF
  delay(j);
  eighthsource(); //LED8 is ON
  delay(i);
  getADCvalue();
  nochannel(); //LED8 is OFF
  delay(j);
}

void sendDataToSerial(char symbol, float data){
  Serial.print(symbol);
  Serial.println(data, 7);
}

void firstsource(){
  digitalWrite(E, LOW);
  digitalWrite(s0, HIGH);
  digitalWrite(s1, LOW);
  digitalWrite(s2, LOW);
  digitalWrite(s3, LOW);
```

```
}

void secondsource(){
  digitalWrite(E, LOW);
  digitalWrite(s0, LOW);
  digitalWrite(s1, HIGH);
  digitalWrite(s2, LOW);
  digitalWrite(s3, LOW);
}

void thirdsource(){
  digitalWrite(E, LOW);
  digitalWrite(s0, LOW);
  digitalWrite(s1, LOW);
  digitalWrite(s2, HIGH);
  digitalWrite(s3, LOW);
}

void fourthsource(){
  digitalWrite(E, LOW);
  digitalWrite(s0, HIGH);
  digitalWrite(s1, LOW);
  digitalWrite(s2, HIGH);
  digitalWrite(s3, LOW);
}

void fifthsource(){
  digitalWrite(E, LOW);
  digitalWrite(s0, LOW);
  digitalWrite(s1, HIGH);
  digitalWrite(s2, HIGH);
  digitalWrite(s3, LOW);
}

void sixthsource(){
  digitalWrite(E, LOW);
  digitalWrite(s0, LOW);
  digitalWrite(s1, LOW);
  digitalWrite(s2, LOW);
  digitalWrite(s3, LOW);
}

void seventhsource(){
  digitalWrite(E, LOW);
  digitalWrite(s0, HIGH);
  digitalWrite(s1, HIGH);
```

```
  digitalWrite(s2, LOW);
  digitalWrite(s3, LOW);
}

void eighthsource(){
  digitalWrite(E, LOW);
  digitalWrite(s0, HIGH);
  digitalWrite(s1, HIGH);
  digitalWrite(s2, HIGH);
  digitalWrite(s3, LOW);
}

void nochannel(){
  digitalWrite(E, HIGH);
}

void getADCvalue() {
  adc0 = ads.readADC_SingleEnded(0);
  Voltage1 = (adc0 * 0.1875)/1000;
  voltages[0] = Voltage1;
  adc1 = ads.readADC_SingleEnded(1);
  Voltage2 = (adc1 * 0.1875)/1000;
  voltages[1] = Voltage2;
  adc2 = ads.readADC_SingleEnded(2);
  Voltage3 = (adc2 * 0.1875)/1000;
  voltages[2] = Voltage3;
  adc3 = ads.readADC_SingleEnded(3);
  Voltage4 = (adc3 * 0.1875)/1000;
  voltages[3] = Voltage4;
  Serial.print(voltages[0]);
  Serial.print(",");
  Serial.print(voltages[1]);
  Serial.print(',');
  Serial.print(voltages[2]);
  Serial.print(",");
  Serial.println(voltages[3]);
  Serial.println();
}
```

Appendix D: **Python Code to Save the Data from Sensors to CSV File**

```
import serial
import time

ser = serial.Serial('COM13', 9600, timeout=1)
time.sleep(1)

logfile = open('test.csv', 'w')

while 1:
        line = ser.readline()
        now = time.strftime("%H:%M:%S", time.localtime())
        a =  "%s, %s" % (now, line)
        print a
        logfile.write(a)
    logfile.flush()
logfile.close()
```

Appendix E: Truth Table and Working Principle of DEMUX CD74HC4067

DEMUX CD74HC4067 works as a digitally controlled analog switch. It switches between sixteen input pins based on the configuration of four sources and enable pin.

Enable pin determines whether the connection between signal pin and channels is open or not. If enable pin is high, it blocks the connection. If enable pin is low, it allows the connection.

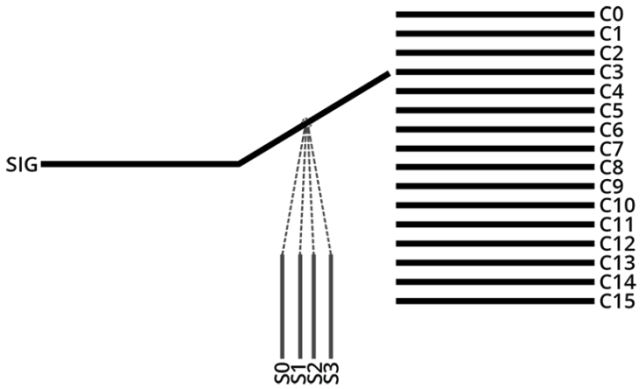
The signal pin (COM pin on the board) connects to channels according to source pins configuration based on the truth table [53]. The truth table and working principle are provided.

In our system, the signal pin connects to ground. The LEDs are connected to VCC. The current flows from VCC to LED, then connected channel of DEMUX. Since the channel connects to signal pin, the current flows to ground through the signal pin and completes the circuit.

S0	S1	S2	S3	$\overline{E}$	SELECTED CHANNEL
X	X	X	X	1	None
0	0	0	0	0	0
1	0	0	0	0	1
0	1	0	0	0	2
1	1	0	0	0	3
0	0	1	0	0	4
1	0	1	0	0	5
0	1	1	0	0	6
1	1	1	0	0	7
0	0	0	1	0	8
1	0	0	1	0	9
0	1	0	1	0	10
1	1	0	1	0	11
0	0	1	1	0	12
1	0	1	1	0	13
0	1	1	1	0	14
1	1	1	1	0	15

H= High Level
L= Low Level
X= Don't Care

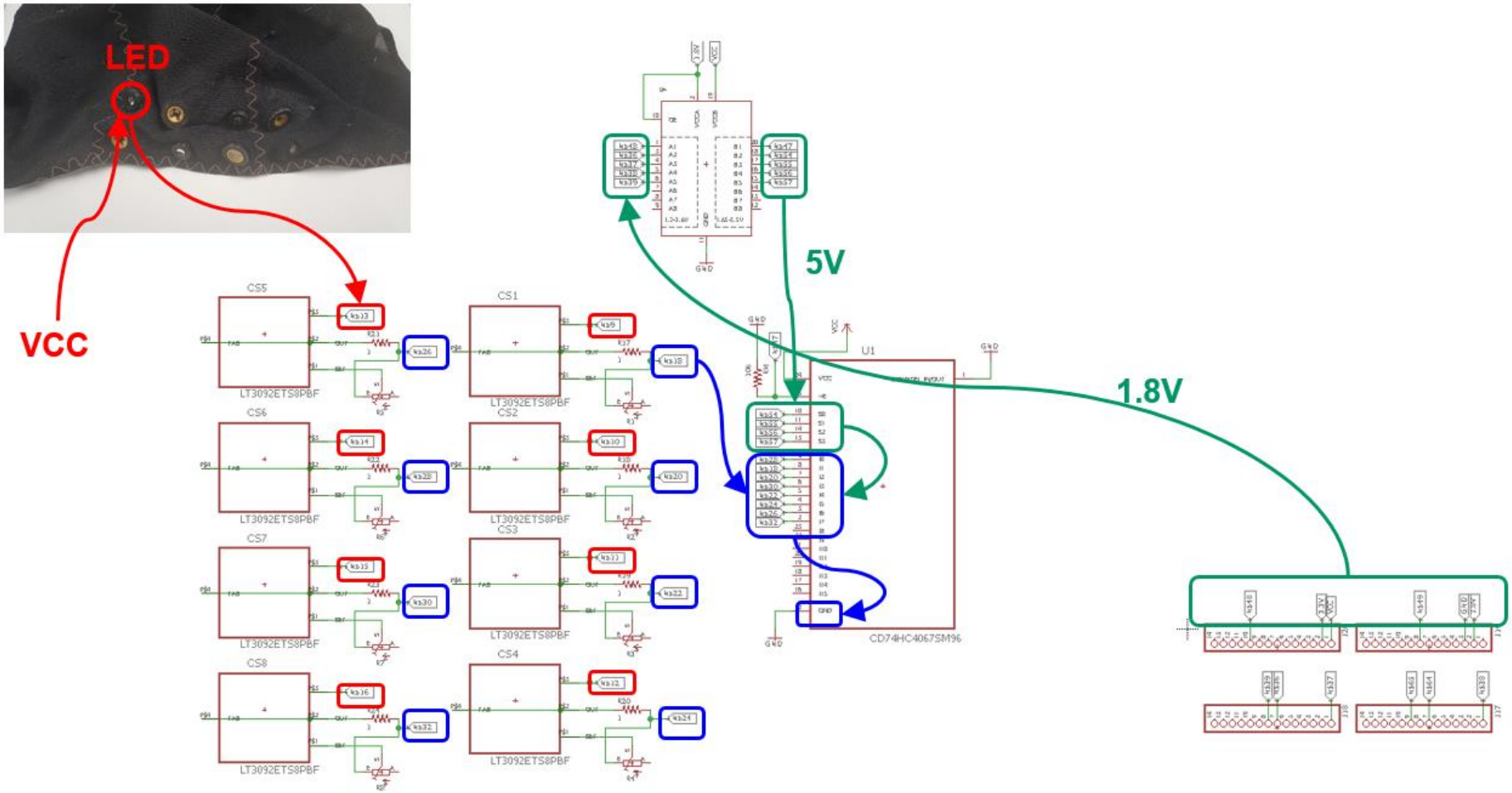
(a)



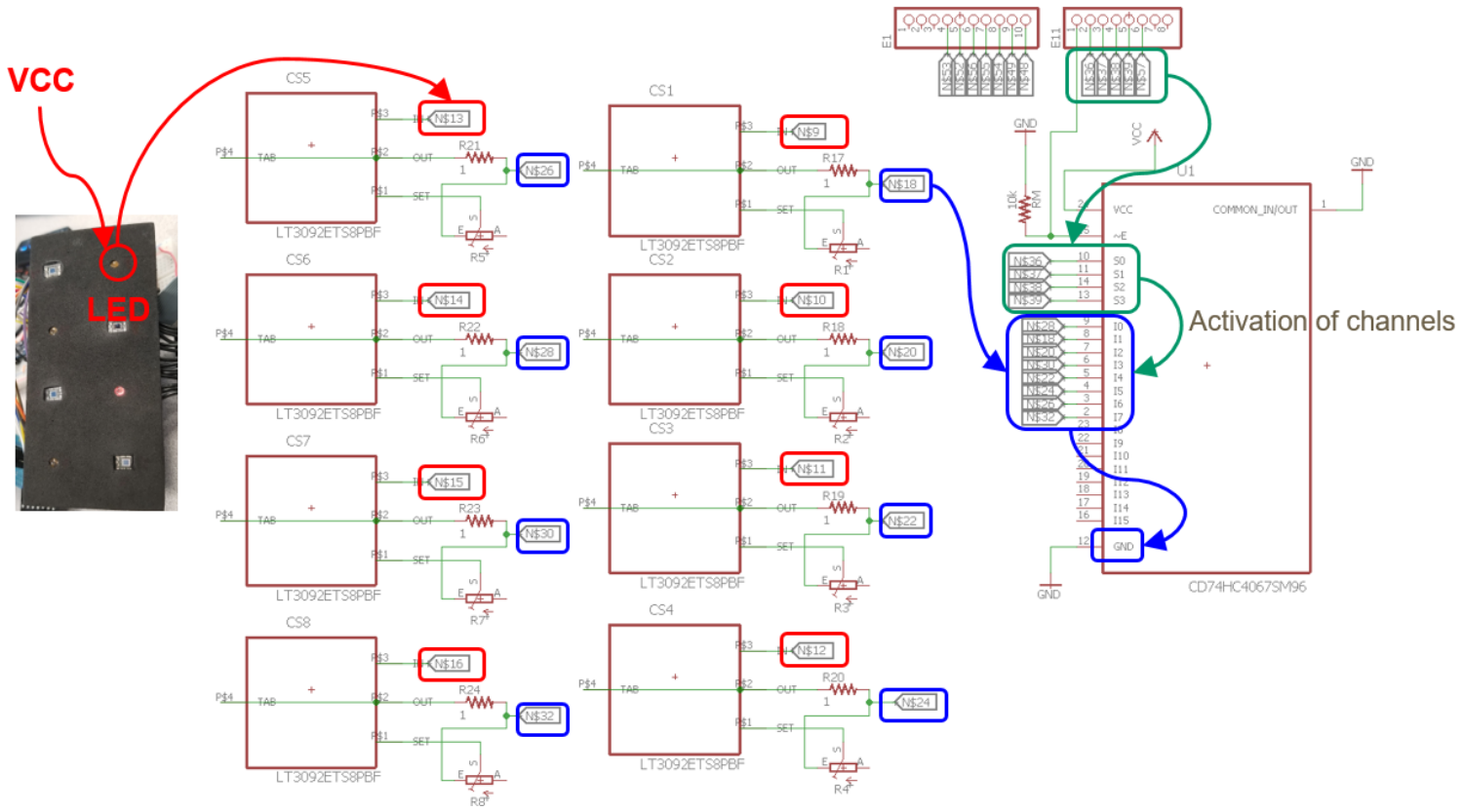
(b)

(a) Function diagram and truth table of DEMUX CD74HC4067, (b) Working principle of DEMUX CD74HC4067

Appendix F: The Schematic of LED Driver Circuit for Version 3

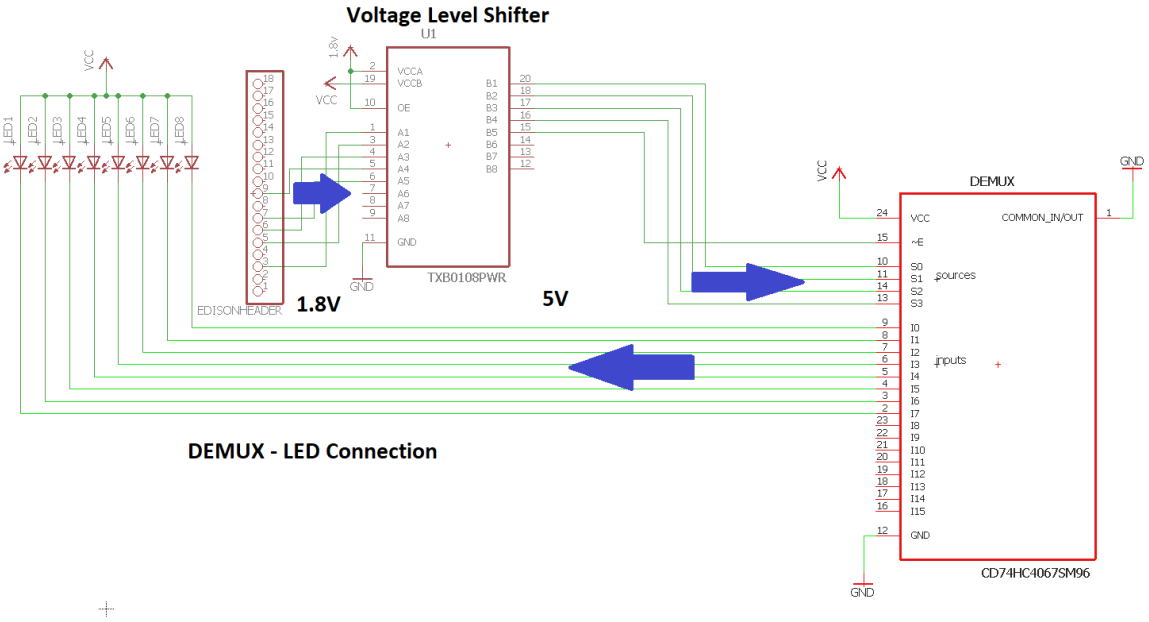


The Schematic of LED Driver Circuit for Version 2

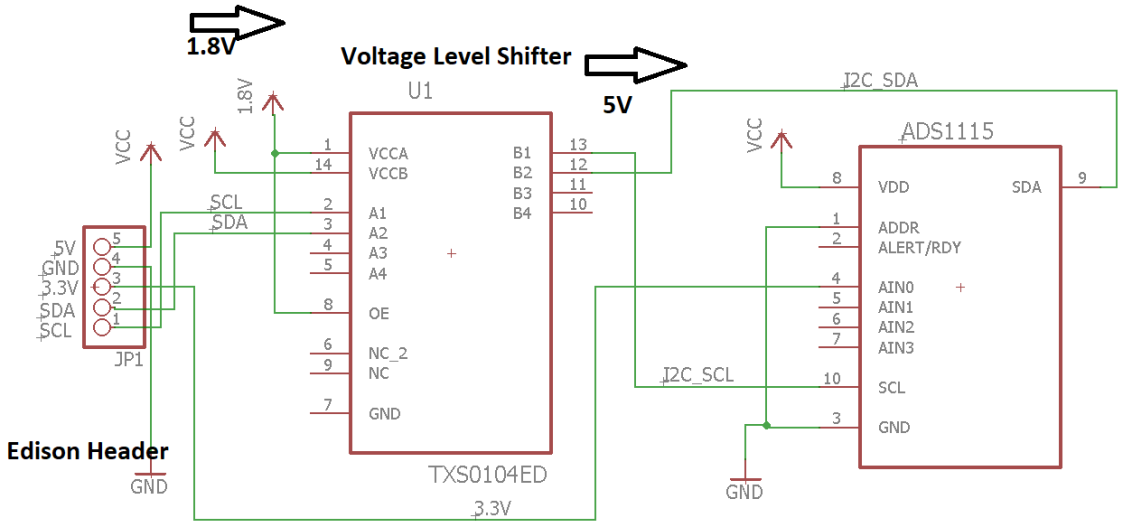


Appendix G: Verification Tests of DEMUX CD74HC4067 and ADS1115 Analog to Digital Converter

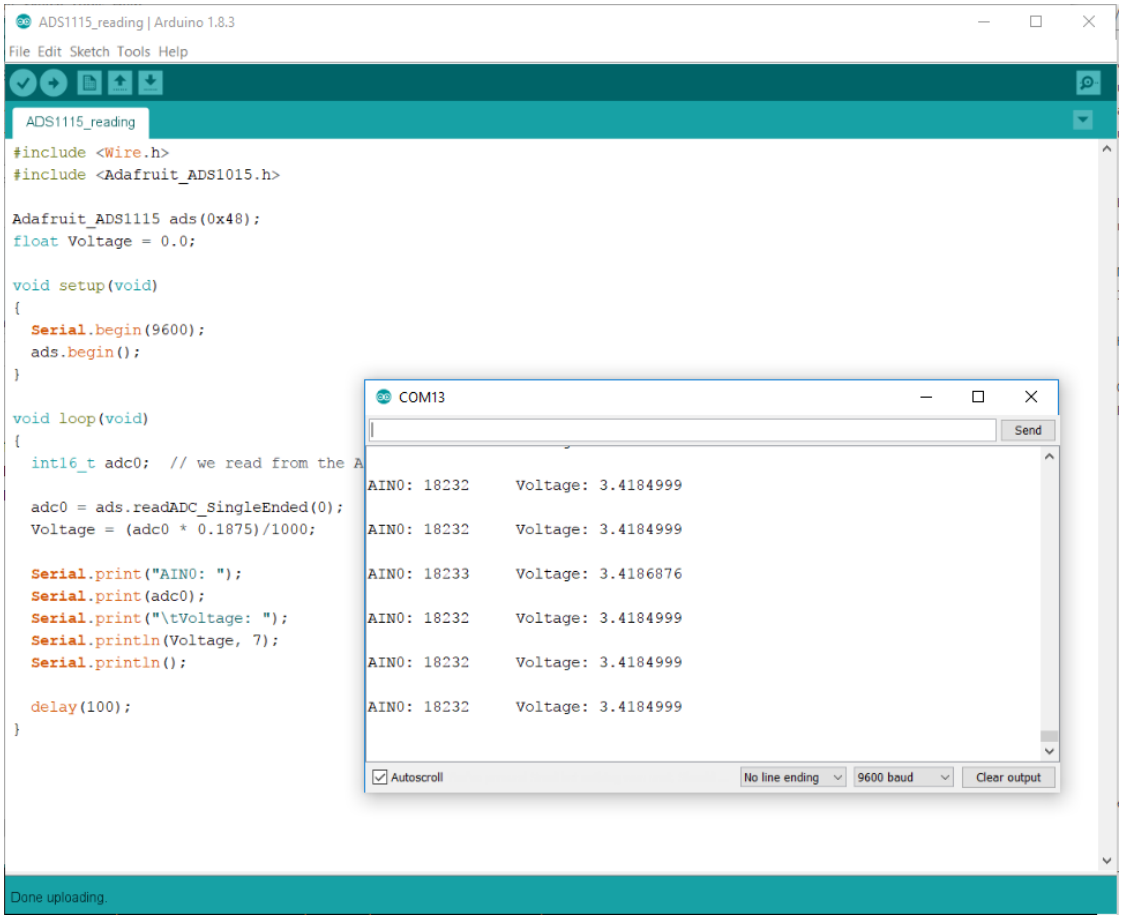
Testing schematic of DEMUX CD74HC4067.



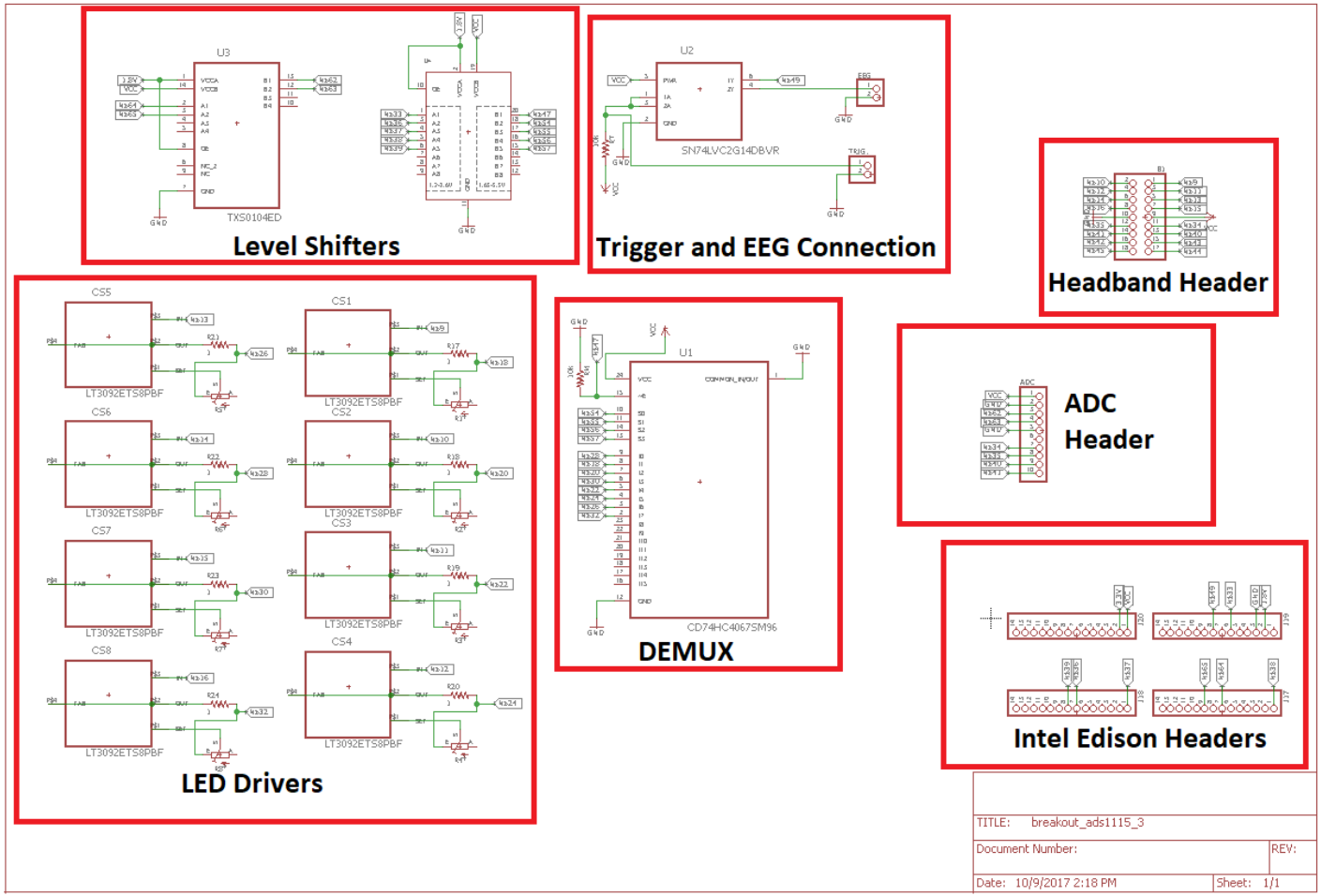
Testing schematic and test results of ADS1115 Analog to Digital Converter.



For validation test, A0 pin of ADS1115 was connected to 3.3V pin of Intel Edison Mini Breakout Board. So 3.4V was seen on the serial monitor as expected.



Appendix H: Schematic and Layout of Implemented System

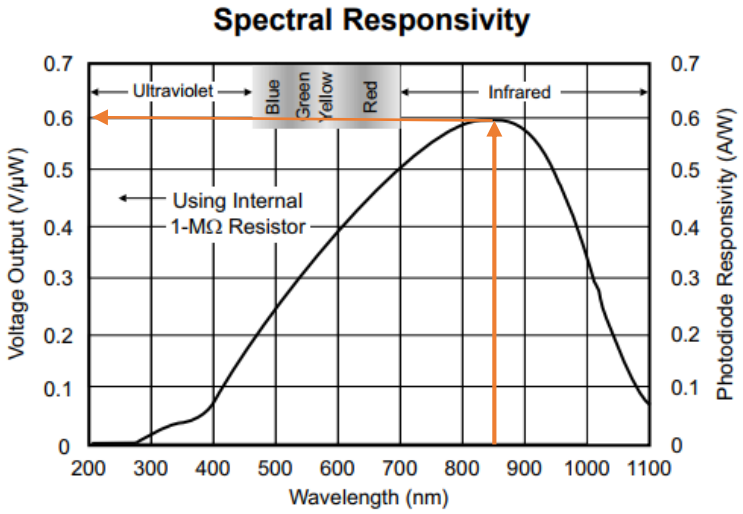


59.99
7.00
28.00
16.99
77.65
ADC
J17
J18
J19
J20
U3
1k
R9
R13
T1
100/220
10
R14
T2
100/220
R24
C28
R17
C23
R18
C27
R12
1k
R11
1k
T4
100/220
R16
R15
100/220
C24
TRIG.
R1
R7
R8

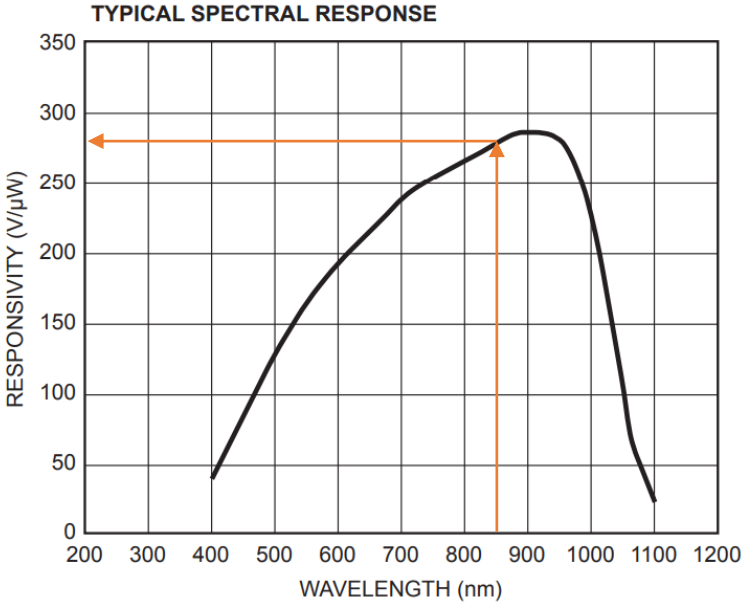
Appendix I: Intel Arduino Breakout Board and Intel Mini Breakout Board Header Pin Comparison

Arduino Header Pin	Breakout Header Pin	Linux GPIO pin	Description	
IO1	J19-8	131	GPIO, UART1 TX output	Trigger
IO2	J17-14	128	GPIO, UART1 clear to send input	DRDY
IO3	J18-7	12	PWM0	
IO4	J18-12	129	UART1 RTS	
IO5	J18-1	13	PWM1	
IO6	J17-1	182	PWM2	
IO7	J19-6	48	GPIO, gyro ready	DEMUX E pin
IO8	J20-6	49	GPIO, gyro interrupt	START
IO9	J18-8	183	PWM3	
IO10	J20-10	41	GPIO, SSP2_FS	CS
	J17-10	111	GPIO, SSP5_FS1	
IO11	J19-11	43	GPIO, SSP2_TXD	DIN
	J17-12	115	GPIO, SSP5_TXD	
IO12	J20-9	42	GPIO, SSP2_RXD	DOUT
	J18-11	114	GPIO, SSP5_RX	
IO13	J19-10	40	GPIO, SSP2 clock output	SCLK
	J17-11	109	GPIO, SSP5 clock output	
IO18	J17-9	28	GPIO, SDA	SDA
IO19	J17-7	27	GPIO, SCL	SCL

Appendix J: Comparison of OPT-101 and ODA-6WB-500M photodetectors' responses



OPT-101



ODA-6WB-500M

BIBLIOGRAPHY

[1] Alzheimer's Association, "2017 Alzheimer's Disease Facts and Figures," *Alzheimers Dement* , vol. 13, pp. 325–373, 2017.

[2] "Statistics on Parkinson's." [Online]. Available: http://www.pdf.org/parkinson_statistics. [Accessed: 01-Jan-2017].

[3] "Who Gets ALS?," 2016. [Online]. Available: http://www.alsa.org/about-als/facts-you-should-know.html?referrer=http://www.ohsu.edu/xd/health/services/brain/in-community/brain-awareness/brain-health/disease-statistics.cfm.

[4] "Living With Brain Injury," 2015. .

[5] WHO, "Neurological disorders public health challenges," *Medicine (Baltimore).*, p. 229, 2006.

[6] S. Sandrone *et al.*, "Weighing brain activity with the balance: Angelo Mosso's original manuscripts come to light," *Brain*, vol. 137, no. 2, pp. 621–633, 2014.

[7] M. Demitri, "Types of Brain Imaging Techniques," 2016. [Online]. Available: https://psychcentral.com/lib/types-of-brain-imaging-techniques/. [Accessed: 01-Jan-2017].

[8] 4-D Neuroimaging, "Basic Principles of Magnetoencephalography." [Online]. Available: http://web.mit.edu/kitmitmeg/whatis.html. [Accessed: 19-Oct-2017].

[9] "Brain Imaging Techniques." [Online]. Available: https://courses.lumenlearning.com/boundless-psychology/chapter/brain-imaging-techniques/. [Accessed: 01-Jan-2017].

[10] “Pros and Cons/Side Effects of PET.” [Online]. Available: http://www.radiology-info.org/nuclear-medicine-positron-emission-tomography/pros-consside-effects.html. [Accessed: 01-Jan-2017].

[11] N. Medicine, “NATIONAL INSTITUTE OF BIOMEDICAL IMAGING AND BIOENGINEERING Nuclear Medicine What is nuclear medicine ? What are radioactive tracers ? What is Single Photon Emission Computed Tomography (SPECT)?,” no. July, 2016.

[12] G. Strangman, D. A. Boas, and J. P. Sutton, “Non-invasive neuroimaging using near-infrared light,” *Biol. Psychiatry*, vol. 52, no. 7, pp. 679–693, 2002.

[13] J. M. Grohol, “What is Functional Near-Infrared Spectroscopy?” [Online]. Available: https://psychcentral.com/lib/what-is-functional-optical-brain-imaging/.

[14] F. Irani, S. M. Platek, S. Bunce, C. Anthony, and D. Chute, “Functional Near Infrared Spectroscopy (fNIRS): An Emerging Neuroimaging Technology with Important Applications for the Study of Brain Disorders,” no. November 2012, pp. 37–41.

[15] M. Ferrari and V. Quaresima, “A brief review on the history of human functional near-infrared spectroscopy (fNIRS) development and fields of application,” *Neuroimage*, vol. 63, no. 2, pp. 921–935, 2012.

[16] L. A. Sordillo, Y. Pu, S. Pratavieira, Y. Budansky, and R. R. Alfano, “Deep optical imaging of tissue using the second and third near-infrared spectral windows.,” *J. Biomed. Opt.*, vol. 19, no. 5, p. 56004, 2014.

[17] A. Meyer-Lindenberg, “From maps to mechanisms through neuroimaging of

schizophrenia," *Nature*, vol. 468, no. 7321, pp. 194–202, 2010.

[18] D. A. Boas and M. Franceschini, "Near infrared imaging," *Scholarpedia*, 2009. [Online]. Available: http://www.scholarpedia.org/article/Near_infrared_imaging. [Accessed: 05-Nov-2017].

[19] F. Jobsis, "Noninvasive, infrared monitoring of cerebral and myocardial oxygen sufficiency and circulatory parameters," *Science (80-.).*, vol. 198, no. 4323, pp. 1264–1267, 1977.

[20] J. S. Wyatt, D. T. Delpy, M. Cope, S. Wray, and E. O. R. Reynolds, "Quantification of Cerebral Oxygenation and Haemodynamics in Sick Newborn Infants By Near Infrared Spectrophotometry," *Lancet*, vol. 328, no. 8515, pp. 1063–1066, 1986.

[21] S. M. Coyle, T. E. Ward, C. M. Markham, E. Ward, and C. M. Markham, "Brain-computer interface using a simplified functional near-infrared spectroscopy system.," *J. Neural Eng.*, vol. 4, no. 3, pp. 219–26, 2007.

[22] S. Lloyd-Fox, A. Blasi, and C. E. Elwell, "Illuminating the developing brain: The past, present and future of functional near infrared spectroscopy," *Neurosci. Biobehav. Rev.*, vol. 34, no. 3, pp. 269–284, 2010.

[23] X. Cui, S. Bray, D. M. Bryant, G. H. Glover, and A. L. Reiss, "A quantitative comparison of NIRS and fMRI across multiple cognitive tasks," vol. 54, no. 4, pp. 2808–2821, 2011.

[24] A. Bozkurt, A. Rosen, H. Rosen, and B. Onaral, "A portable near infrared spectroscopy system for bedside monitoring of newborn brain.," *Biomed. Eng.*

Online, vol. 4, no. 1, p. 29, 2005.

[25] G. Yurtsever, H. Ayaz, F. Kepics, B. Onaral, and K. Pourrezaei, "Wireless, Continuous Wave Near Infrared Spectroscopy System for Monitoring Brain Activity," no. April 2014, 2006.

[26] H. Atsumori *et al.*, "Development of a multi-channel, portable optical topography system," *Annu. Int. Conf. IEEE Eng. Med. Biol. - Proc.*, pp. 3362–3364, 2007.

[27] H. Atsumori, "Noninvasive imaging of prefrontal activation during attention-demanding tasks performed while walking using a wearable optical topography system," *J. Biomed. Opt.*, vol. 15, no. 4, p. 46002, 2010.

[28] S. K. Piper *et al.*, "A wearable multi-channel fNIRS system for brain imaging in freely moving subjects," *Neuroimage*, vol. 85, pp. 64–71, 2014.

[29] C. Zysset *et al.*, "Textile integrated sensors and actuators for near-infrared spectroscopy.," *Opt. Express*, vol. 21, no. 3, pp. 3213–24, 2013.

[30] P. Pinti *et al.*, "Using Fiberless, Wearable fNIRS to Monitor Brain Activity in Real-world Cognitive Tasks," *J. Vis. Exp.*, no. 106, pp. 1–13, 2015.

[31] D. Agro *et al.*, "Applications in electronics pervading industry, environment and society: Applepies 2016," *Lect. Notes Electr. Eng.*, vol. 429, no. January, 2017.

[32] M. Rodriguez, "Development of a versatile wireless fNIR system," 2011.

[33] M. Abtahi, G. Cay, M. J. Saikia, and K. Mankodiya, "Designing and Testing a Wearable , Wireless fNIRS Patch *," 2016, pp. 6298–6301.

[34] M. Cope, "The Application Of Near Infrared Spectroscopy To Non Invasive

Monitoring Of Cerebral Oxygenation In The Newborn Infant," 1991.

[35] A. Bozkurt and B. Onaral, "Safety assessment of near infrared light emitting diodes for diffuse optical measurements.," *Biomed. Eng. Online*, vol. 3, no. 1, p. 9, 2004.

[36] A. von Lühmann, "openNIRS Documentation," Karlsruhe Institute of Technology, 2014.

[37] "LT3092." [Online]. Available: http://www.linear.com/product/LT3092. [Accessed: 01-Jan-2017].

[38] "CD74HC4067." [Online]. Available: http://www.ti.com/product/CD74HC4067. [Accessed: 01-Jan-2017].

[39] "ADS1258." [Online]. Available: http://www.ti.com/product/ADS1258. [Accessed: 01-Jan-2017].

[40] "Intel Edison." .

[41] "Arduino IDE." [Online]. Available: https://www.arduino.cc/en/Main/Software. [Accessed: 01-Jan-2017].

[42] "MATLAB R2015a." [Online]. Available: https://www.mathworks.com/products/new_products/release2015a.html.

[43] "ODA-6WB-500M," no. 805, pp. 499–501.

[44] "ADS1115," 2016.

[45] "Yocto Project," 2016. [Online]. Available: https://www.yoctoproject.org/about. [Accessed: 01-Jan-2017].

[46] "libmraa." [Online]. Available: https://iotdk.intel.com/docs/master/mraa/. [Accessed: 01-Jan-2017].

[47] E. R. Laskowski, “What’s a normal resting heart rate?,” 2015. [Online]. Available: https://www.mayoclinic.org/healthy-lifestyle/fitness/expert-answers/heart-rate/faq-20057979. [Accessed: 01-Jan-2017].

[48] “Dictionary.” [Online]. Available: https://www.google.com/search?rlz=1C1CHBF_enUS770US770&q=Dictionary#dobs=occlusion.

[49] D. Wyser, O. Lambercy, F. Scholkmann, M. Wolf, and R. Gassert, “Wearable and modular functional near-infrared spectroscopy instrument with multidistance measurements at four wavelengths,” *Neurophotonics*, vol. 4, no. 4, p. 1, 2017.

[50] Nydal Dahl, “PNP Transistor – How Does It Work?,” 2016. [Online]. Available: https://www.build-electronic-circuits.com/pnp-transistor/. [Accessed: 01-Jan-2017].

[51] Nydal Dahl, “How Transistors Work – A Simple Explanation,” 2014. [Online]. Available: https://www.build-electronic-circuits.com/how-transistors-work/. [Accessed: 01-Jan-2017].

[52] J. List, “Intel Discontonues Joule, Galileo, and Edison Product Lines,” 2017. [Online]. Available: https://hackaday.com/2017/06/19/intel-discontinues-joule-galileo-and-edison-product-lines/. [Accessed: 01-Jan-2017].

[53] O. Information, “CD74HCT4067 High-Speed CMOS Logic,” no. February 1998, 2003.